

68

\$2.95_{USA}

Australia
Singapore
Malaysia

A \$ 4.75
S \$ 9.45
M \$ 9.45

New Zealand
Hong Kong
Sweden

NZ \$ 4.75
H \$23.50
30:-SEK

MICRO JOURNAL

VOLUME IV ISSUE IX • Devoted to the 68XX User • September 1982
"Small Computers Doing Big Things"

SERVING THE 68XX USER WORLDWIDE

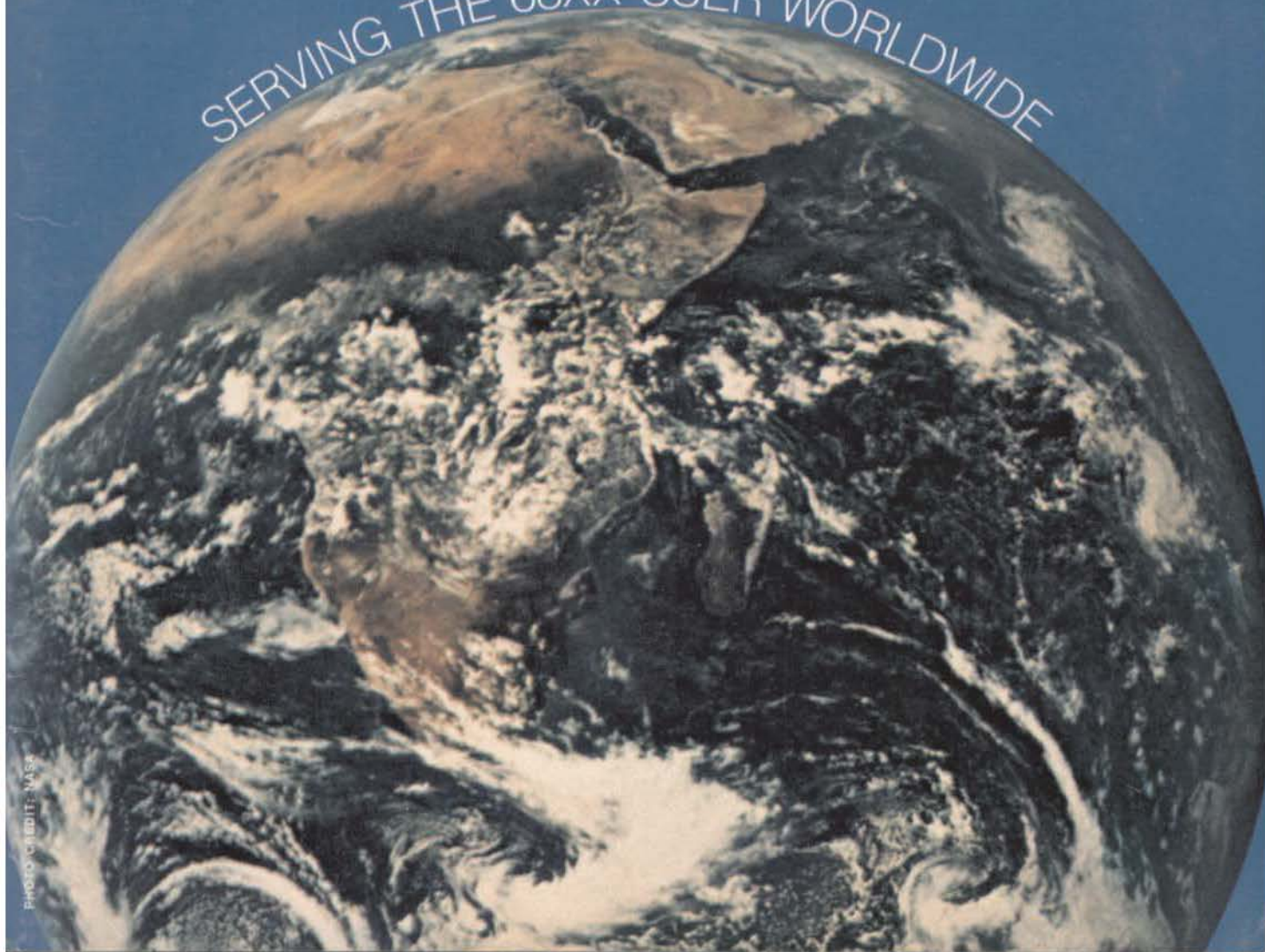


PHOTO CREDIT: NASA



YOUR CHOICE-smart either way

- Over 140 software driven functions
- 82 x 24 or 82 x 20 screen format — software selectable
- High resolution 7 x 12 matrix characters — P-31 green phosphor
- Upper/lower case character set — plus graphics character set
- 56-key alphanumeric keyboard — plus 12-key cursor, numeric pad
- Internal editing functions — insert, delete, scroll, roll, slide, etc.
- Parallel printer I/O port
- 50 to 38,400 baud operation — programmable
- Cursor type, cursor position, print control characters, protected fields, shift inversion, dual intensity and many other features

8212 — twelve-inch diagonal screen or 8209 — nine-inch diagonal screen



SOUTHWEST TECHNICAL PRODUCTS CORPORATION
219 W. RHAPSODY
SAN ANTONIO, TEXAS 78216 (512) 344-0241

Pascal for 6809

Pascal for the 6809 is a true native code compiler. Unlike the usual P-code Pascals which run in an interpretive manner, ours produces efficient assembly language mnemonics which can be assembled and run directly. This compiler is available for both 6809 FLEX™ and UniFLEX™. Many features not found in other Pascal systems were implemented while avoiding those features completely non-standard. Features of the Pascal system include:

- Supports most of Jensen and Wirth specification
- Produces fast and efficient 6809, native code
- FLEX run-time package may be trimmed
- Double precision real numbers (16.8 digits)
- Implements scalar, subrange and structured data types
- Standard I/O using file buffer pointers
- Dynamic storage allocation
- Ability to call other Pascal programs
- FLEX version may call assembly language programs
- Buffered or single character terminal input
- Standard math functions: SIN, COS, ARCTAN, EXP, LN, SQR, SQRT
- Random number generator function
- Many usable, sample programs included
- UniFLEX version supports:
 - Random file positioning
 - Ability to call various UniFLEX system routines
 - Ability to execute UniFLEX utility commands

Pascal on diskette for 5" and 8" 6809 FLEX is available for \$200.00. The 5" version requires two disk drives. The UniFLEX version is \$300.00 and includes one year of maintenance. All orders should include 3 percent for postage and handling (10 percent on foreign orders).

™FLEX and UniFLEX are trademarks of Technical Systems Consultants, Inc.

 **technical systems
consultants, inc.**
111 Providence Rd., Chapel Hill, N.C. 27514
(919) 493-1451

'68'

Portions of text prepared using the following.

SWTPC 6800-6809-OMAF2-CDS1-CT82-Sprint 3
Southwest Technical Products
219 W. Rhapsody
San Antonio, Texas 78216

EDITOR - WORD PROCESSOR
Technical Systems Consultants, Inc.
Box 2573, W. Lafayette, IN 47906
FLEX is TM of TSC

GIMIX Super Mainframe-Assorted memory boards
GIMIX Inc.
1337 West 37th Place
Chicago, IL 60609

Editorial Staff:

Don Williams Sr., Publisher

Larry E. Williams, Executive Editor

Tom E. Williams, Production Editor

Robert (Bob) Nay, Color Computer
Editor

Subscriptions and Office Manager
Mary Robertson

Accounting Office manager
Joyce Williams

Contributing Editors:

Ron Anderson

Ray Cadmus

Norm Cammo

Dr. Theo Elbert

Dr. E.M. 'Bud' Pass

Special Technical Projects:

Clay Abrams K6AEP

Tom Hunt

TABLE OF CONTENTS

Vol. IV, Issue IX

September '82

FLEX User Notes	8	Anderson
COLOR User Notes	11	Nay
Color Clinic	13	DI Stefano
"C" User Notes	14	Cammo
TELECON "C" Review	18	Urle
Simple Winchester Interface	20	Zeff
SPEAK & SING Review	21	Staff
6800 to 6809	24	Pass
Home Acct Prog	26	Watson/ Brady
ET3400 BASIC	29	Wolach
Bit Bucket	31	
Classifieds	41	

MICRO JOURNAL

Send All Correspondence To:

Computer Publishing Center
68 MICRO JOURNAL
5900 Cassandra Smith
PO Box 849
Hixson, TN 37343
615 842-4600

Copyrighted 1982 by Computer Publishing, Inc. (CPI)

68 Micro Journal is published 12 times a year by Computer Publishing, Inc. Second class postage paid at Hixson, Tennessee and additional entries. Postmaster: send Form 3579 to 68 Micro Journal, PO Box 849, Hixson, Tennessee.

SUBSCRIPTION RATES

USA

1-Year \$24.50 2-Years \$42.50 3-Years \$64.50

FOREIGN

See Page 52

Items Submitted for Publication

Articles submitted for publication should be accompanied by the authors full name, address, date and telephone number. It is preferred that articles be submitted on either 5 or 8 inch diskette in TSC Editor format or STYLO format. All diskettes will be returned.

The following TSC Text Processor commands ONLY should be used (due to our proportional processor): .sp space, .pp paragraph, .fl fill and .nf no fill. Also please do not format within the text with multiple spaces. The rest we will enter at time of editing.

STYLO commands are all acceptable except the .pg page command, we print edited text files in continuous text.

All articles submitted on diskettes should be in TSC FLEX* format, either FLEX2 6800, or FLEX9 6809 any version.

If articles are submitted on paper they should be on white 8X11 bond or better grade paper. No hand written articles (hand written or drawn art accepted). All paper submitted articles will be photo reproduced. This requires that they be typed or produced with a dark ribbon (no blue), single spaced and type font no smaller than 'elite' or 12 pitch. Typed text should be approximately 7 inches wide (will be reduced to column width of 3 1/2 inches). Please use a dark ribbon!

All letters to the editor should also comply with the above and bear a signature. Letters of 'gripes' as well as 'praise' are solicited. We attempt to publish all letters to the editor verbatim, however, we reserve the right to reject any submission for lack of 'good taste'. We reserve the right to define what constitutes 'good taste'.

Advertising: Commercial advertisers please contact the 68 Micro Journal advertising department for current rate sheet and requirements.

Classified: All classified must be non-commercial. Maximum 20 words per classified ad. Those consisting of more than 20 words should be figured at .35 cents per word. 20 words or less \$7.50 minimum, one time, paid in advance. No classified ads accepted by telephone.



2MHZ 6809 SYSTEMS

GIMIX offers you a variety to choose from!

38 MB WINCHESTER SYSTEM \$17,498.99

HARDWARE FEATURES:

- ★ 2MHz 6809 CPU
- ★ 512KB Static RAM
- ★ 8 RS232C Serial Ports
- ★ 2 Parallel Ports
- ★ DMA Double Density Floppy Disk Controller
- ★ Dual 8" DSDD Floppy Disk System
- ★ Dual Winchester Subsystem with Two 19 MB 5 1/4" Winchester Drives

SOFTWARE FEATURES:

- ★ OS-9 LEVEL TWO Multi-User Operating System
- ★ OS-9 Debugger
- ★ OS-9 Text Editor
- ★ OS-9 Assembler

19 MB WINCHESTER SYSTEM \$8998.09

HARDWARE FEATURES:

- ★ 128K Static Ram
- ★ 2MHz 6809 CPU
- ★ 19 MB 5 1/4" Winchester DMA Subsystem
- ★ 4 RS232C Serial Ports
- ★ 1 MB 5 1/4" Floppy Disk Drive
- ★ DMA Double Density Floppy Disk Controller

SOFTWARE FEATURES:

- ★ OS-9 LEVEL TWO Multi-User Operating System
- ★ OS-9 Text Editor
- ★ OS-9 Debugger
- ★ OS-9 Assembler

128KB MULTI-USER SYSTEM \$6997.39

HARDWARE FEATURES:

- ★ 2MHz 6809 CPU
- ★ DMA Double Density Floppy Disk Controller
- ★ 128KB Static Ram
- ★ 2 RS232C Serial Ports
- ★ Dual 8" DSDD Floppy Disk System

SOFTWARE FEATURES: Your choice of either UniFLEX or OS-9 LEVEL TWO. Both are Unix-like Multi-User/Multi-Tasking Operating Systems.

56KB FLEX / OS-9 "SWITCHING" SYSTEM \$4148.49

HARDWARE FEATURES:

- ★ 2MHz 6809 CPU
- ★ 56K Static Ram
- ★ 2 RS232C Serial Ports
- ★ DMA Double Density Floppy Disk Controller
- ★ 2 Built-in 5 1/4" 40tr DSDD Disk Drives (80 Track DSDD Drive Option .. add \$400.00)

SOFTWARE FEATURES:

- ★ GMXBUG monitor — FLEX Disk Operating System
- ★ OS-9 LEVEL ONE Multi-tasking operating system for up to 56K of memory

WINCHESTER SUBSYSTEMS

Winchester packages are available for upgrading current GIMIX 6809 systems equipped with DMA controllers, at least one floppy disk drive, and running FLEX, OS-9 LEVEL ONE or OS-9 LEVEL TWO. The packages include one or two 19MB (unformatted) Winchester drives, DMA Hard Disk Interface, and the appropriate software drivers. The Interface can handle two 5 1/4" Winchester Drives, providing Automatic Data Error Detection and Correction; up to 22 bit burst error detection and 11 bit burst error correction. **UniFLEX NOW AVAILABLE**

Dual drives can be used together to provide over 30 MBytes of on line storage .. or use one for back-up of the other. (More convenient and reliable than tape backup systems.)

#90 includes one 19MB Drive, Interface, and Software \$4288.90

#91 includes two 19MB Drives, Interface, and Software \$6688.91

Contact GIMIX for systems customized to your needs or for more information.

50 HZ Export Versions Available

GIMIX Inc. reserves the right to change pricing and product specifications at any time without further notice.

GMX is a trademark of GIMIX Inc.

GIMIX and GHOST are registered trademarks of GIMIX Inc.

FLEX and UniFLEX are trademarks of Technical Systems Consultants Inc.

OS-9 is a trademark of Microware Inc.

1337 WEST 37th PLACE
CHICAGO, ILLINOIS 60609

(312) 927-5510

TWX 910-221-4055

GIMIX INC.

1982 GIMIX Inc.

Build performance into your system with OS-9[™] software tools

Unix[®]-based, multitasking, modular, and versatile; these key features are some of the reasons why more 6809 computer manufacturers have selected OS-9 as their standard operating system than any other. And OS-9 has been put to work by thousands of users in almost every conceivable computer application in business, science, industry, education, and government.

Your operating system should not be a barrier between you and your computer. OS-9 is very friendly and easy to use. Its modular structure makes it easy to customize, plus its comprehensive documentation shows you exactly how to interface it to just about any I/O device.

OS-9's advanced features unleash the performance potential of almost any 6809 computer — large or small. In many respects the OS-9/6809 combination is more powerful than many *minicomputers*!

There are two basic versions of OS-9. Both have the same basic features and capabilities. OS-9 Level One runs on small to medium sized systems having up to 64K memory. The Level Two version runs on medium to large size systems having memory management hardware and up to 1 megabyte of memory, and includes record and file locking for multiuser database applications.

Here are just a few reasons why you should insist on OS-9 for your microcomputer system.

- Over 40 utility commands
- Friendly "Shell" command interpreter
- Tree-structured multilevel file directories

- Full timesharing support with log-in and file security
- Fast, secure random and sequential access files
- Comprehensive English language error messages
- Compact real-time multitasking executive
- Hardware or software memory management
- Device independent interrupt-driven I/O
- Fully ROMable for small control systems
- Standard versions available from manufacturers of most popular 6809 computers

OS-9 PASCAL Language Compiler

- most complete and versatile PASCAL available for the 6809
- capable of generating P-code for interpretive execution while debugging OR
- highly optimized 6809 assembly language source code output for maximum speed
- "virtual memory" P-code interpreter lets you run large PASCAL programs

CIS COBOL * Compiler**

- ideal for most demanding business applications
- features ISAM, Debug, ACCEPT/DISPLAY and Interprogram Communications modules
- retains full compatibility with CP/M software
- meets ANSI 1974 Level One COBOL standard and is GSA certified
- Also available FORMS 2 automatic program generator for easy interactive design of screen oriented applications.

BASIC09[™] Structured Basic Interactive Compiler

- fastest and most comprehensive full Basic language available for the 6809
- combines standard Basic with the best features of PASCAL
- features compiler speed, interpreter friendliness and superlative debugging facilities
- option available: Run Basic ROMable run-time system for compiled Basic 09

C Language Compiler

- complete implementation of the UNIX version 7 C language
- includes INT, CHAR, SIGNED, UNSIGNED, FLOAT AND LONG data types, structures, unions, standard C library and a full preprocessor with macro definitions
- generates fully reentrant 6809 assembly language source code output

For more information contact your computer supplier or



MICROWARE

Microware Systems Corporation
5835 Grand Avenue, Des Moines,
Iowa 50312 515-279-8844 • Telex
910-520-2535

*Unix is a trademark of Bell Laboratories. ***CIS Cobol is a trademark of Micro Focus, Inc. OS-9[™] and Basic09 are trademarks of Microware and Motorola, Inc.

LEVEL TWO

Expand your 6809 computer to a fast, efficient multi-user system utilizing up to one megabyte of memory, almost any I/O device, and comprehensive implementations of the most-wanted programming languages: Basic[®], C, Pascal, Cobol, and Assembler.

With OS-9[™] Level Two, your computer is transformed into the ultimate software development system with performance and features found only on large and costly computing systems. It brings to your fingertips the friendliness and power of a Unix[®]-style environment.

As a multi-user system...

OS-9 Level Two excels with a multi-level directory system, a fast random access file system with record lockout, user name/password logon protection, "pipes" for interprogram communication, and full file security. The versatile "Shell" command interpreter makes it easy for each user to run interactive or multiple background programs with I/O redirection to or from any file or I/O device.

As a real-time system...

Your OS-9 based computer can sense, monitor, control, and communicate with the real world thanks to OS-9's highly modular and user expandable structure. Adding customized I/O to OS-9 is almost too easy: software interfaces are simple, modular, and well documented. That is why OS-9 users routinely interface it to almost every kind of peripheral device and instrumentation.

For large systems...

OS-9 Level Two can handle over one megabyte of RAM and ROM memory with extraordinary efficiency. When, as is often the case, two or more tasks run the same program (such as Basic[®]) they automatically "share" just one copy in memory. Also, OS-9 Level Two typically resides in less than 24K memory. Savings like these give OS-9 based systems large capacity without having to resort to performance-robbing techniques such as disk swapping.

OS-9 Level Two is available exclusively from manufacturers of most popular 6809 computers equipped with memory management hardware. They offer versions specifically tailored to their computers for use with both new and existing systems.

For more information about OS-9 Level Two contact your computer supplier, or



MICROWARE

Systems Corporation
5835 Grand Avenue, Des Moines,
Iowa 50312
phone 515-279-8844
Telex 910-520-2535

* Unix is a trademark of Bell Laboratories.
** OS-9 and Basic[®] are trademarks of
Microware and Motorola, Inc.

DynaStar

WORD PROCESSING SYSTEM FOR OS-9

OS-9 USERS:

If your computer has a SCREEN and you're still struggling with an editor that only knows about LINES, then obviously YOU don't know about

DynaStar

DynaStar is a powerful, menu-driven screen editor equally suited to the tasks of program preparation and document processing. With the addition of the optional DynaForm print formatter, it is the best word-processing package you can buy for your OS-9 system.

DynaStar Version II is now available and features no nonsense "what you see is what you get" editing for virtually any terminal with or without cursor addressing (it must be at least able to go to "home"). To edit, simply place the cursor where you want it, and type. Any printable character you type is entered directly into your text, and any non-printable control character causes immediate execution of an editing command. Single keystroke commands permit movement of the cursor in any direction, by character, tab, word, line, or screen full, and deletion of characters, words (left or right) or a whole line. Two keystroke commands augment this set by moving the cursor to the left margin, top or bottom of the screen, beginning or end of the edit buffer, or the beginning of the next paragraph. You can search for any string, replace with any other, do it again, mark original blocks of text, copy, move or delete blocks, read or write to side files, set tabs and margins, or center the current line.

DynaStar features automatic word wrap and it can right justify text as you enter it so you will see exactly how it will look before you print it. If you later make alterations or change the margins, you can reform the text a paragraph at a time with two keystrokes. For programmers, there is a special automatic indent mode to help you write well-structured code. DynaStar includes a Shell command which lets you do almost anything (including edit another file) without even losing your place in your current document, and it permits editing of large disk files in stages without forcing you to break up your files.

If you want to define more powerful commands, DynaStar includes a macro facility which lets you convert any control character to one or a string of characters of your choice. You can use this feature to create global search and replace commands, insert "boilerplate" or simply re-map your keyboard. You can also provide a special "startup string" which is automatically executed whenever you enter the editor to set up modes such as auto-justification, display a directory, define your favorite macros, or re-map the keyboard.

For complete word processing, we offer our DynaForm text formatter which provides all the standard features such as pagination, headers and footers with page numbers, single space, double space, multiple space, bold face, double strike, and underline. DynaForm has its own macro facility with string variables, nested include files, a full merge-print capability for generating form letters and mailing lists, and it can generate an index automatically, sorted alphabetically or by page number. You can call it from DynaStar to proof-print the active edit buffer, or by itself to print a disk file while you edit another.

DynaStar II \$149.95
DynaForm text formatter \$149.95
Both purchased together \$279.90
Note: DynaStar Version II (no macros) will be available at the original price until May 31, and current owners may upgrade to Version II with full credit until June 30.

AVAILABLE SOON FOR FLEX 9

Spelltest

From Dale Puckett

FOR OS-9 OR FLEX

SPELLTEST is the most versatile 68XX spelling checker available. MENU'S MAKE SPELLING EASY. From the menu you may print a list of valid words, or a list of words you've entered. Read your text, still in the editor, and you'll see additional words you've entered. While in the editor, you can also use the dictionary to find the correct spelling of a word. Design your own dictionary. at home in the cost of 22,000 words. The America English word list is included. 500 built-in words, and 300 specific to your field. The text and allows a large file to be processed even in small computers.

PRICE \$199.00

A/BASIC

Basic Compiler

For OS-9 or FLEX

If you are still programming in assembler, this is the program for you! This BASIC compiler generates pure, fast efficient 6809 machine code from easy to write BASIC source programs. Uses ultra-fast integer math, extended string functions, boolean operators and run-time operations. Output is ROMable and RUNS WITH-OUT ANY RUN-TIME PACKAGE. Supports IF-THEN-ELSE structure, random access, and several improvements over the original 6809 version sold by Microware. Optimized for the 6809, A/BASIC is 8 to 10 times faster than the original 6800 version, and produces code approximately 30% smaller.

SPECIAL

CHESSE program coded in A/BASIC (originally sold for \$50) is included FREE on the disk in both source and object for your enjoyment. Also some utilities are included for testing and examples, all in source on the disk!

ONLY \$150.00
specify OS-9 or FLEX

PLOT

Now you can have GRAPHICS added to all your programs. Just write the data out to a virtual array and call PLOT. PLOT is written in TSC X BASIC and the source is included on the disk. INFINITE RESOLUTION GRAPHICS ON YOUR TERMINAL OR PRINTER. HISTOGRAMS, BAROGRAPHS, XY PLOTS PLUS OTHERS. IN TSC X BASIC. SOURCE INCLUDED ON DISK. \$44.95

TOOLKIT NO1

The Basic Programmers Toolkit
by Dick Bartholomew

The Basic Programmers Toolkit gives the BASIC programmer the power and flexibility never before achieved under FLEX.

PRICE \$49.95 object only
\$69.95 with source on disk!

TOOLKIT NO2

The Programmers Toolkit
by Dick Bartholomew

The Programmers Toolkit is a package of utilities and programs that extend the capabilities of FLEX to the utmost.

PRICE \$49.95 object only
\$69.95 with source on disk!

Dynasoft RAVV 1.1 for OS-9

Dynasoft Pascal

FLEX version 1.1

Chain, Read, Write, Fork, Send, Wait, Setstatus, SetPrio, Kill, and fast but powerful editor.

Object Only \$69.95

Add for run-time source on disk \$30.00

Add for source of Dynasoft Pascal itself \$125.00

Version 1.5 Now Available

40% Faster

15 new features of the enhancements:

Close, Delete, Getstatus, This is an excellent set of utilities

to write a Star

CRASMB

MULTI CPU CROSS ASSEMBLER FOR 6809
FLEX OR OS-9
by Frank Hoffman

CRASMB is a conditional macro assembler with the capability to use different CPU overlays in order to cross assemble. These CPU overlays are called "CPU PERSONALITY MODULES" (CPM's) and can be called from a source file, thereby making it easy to create object code for a variety of CPUs. It is also possible to create new CPM's yourself for any 8 or 16 bit CPU. The information needed is included in the manual. If you decide to do this, it would be advisable to purchase the source for one of the CPM's and modify it rather than starting from scratch. CPM's are currently available for the following CPUs: 6809, 6800, 6805, 6502, Z80, 8080, 1802, and others coming. FLEX \$39.95 with any CPM OS-9 \$200.00 with 6809 CPM CPM's \$25.00 each 35.00 each CPM source \$25.00 each 35.00 each Specify FLEX or OS-9 when ordering

THE BILL PAYER SYSTEM™

THE BILL PAYER is a package of 10 menu driven programs in TSC Extended Basic. This powerful system helps you keep track of your bills. You can create a vendor list, enter invoices to be paid, generate reports about them, print your checks and much more. Uses random access files.

Explore Package now included at the same price. THE PURCHASE ORDER system adds purchase orders to the BILL PAYER. This package of programs adds another level of control to your expenditures. Prints out purchase orders and keeps track of purchases. Requires the Bill Payer to work.

INCOME/EXPENSE LEDGER. This valuable package is most appreciated at tax time. Allows up to 99 income and expense numbers. Ties into the PURCHASE ORDER system, and the Bill Payer. Includes manual and source supplied on disk in TSC Extended Basic.

THE BILL PAYER
PURCHASE ORDER
INCOME/EXPENSE

\$11.95 each \$29.95

COLOR COMPUTER USERS

**FHL COLOR FLEX, THE MOST POPULAR DOS FOR THE 6809
FROM THE LARGEST SUPPLIER OF FLEX SOFTWARE IN THE WORLD!**

Now you can run FLEX, OS-9 and Radio Shack disk software on your Color Computer. If you have a 32K Color Computer with the Radio Shack disk system, all you need to do is make a trivial modification to access the hidden 32K, as described in the Feb. issue of COLOR COMPUTER NEWS and the April issue of 68 Micro. You can get FLEX from us right now. OS-9 will be ready by summer. Please note that this will only work with the Radio Shack disk system and 32K/48K memory chips that RS calls 32K. Maybe they put 64K's in yours, too. If you don't have a copy of the article, send a legal size SASE (40¢ stamps) and we'll send it to you.

Using this system to run FLEX AND OS-9 has many advantages. First, it gives you 48K from zero right up to FLEX. This means that ALL FLEX compatible software will run with NO MODIFICATIONS and NO PATCHES! There are no memory conflicts because we moved the screen up above FLEX which leaves the lower 48K free for user programs.

What you end up with is 48K for user programs, 8K for FLEX and another 8K above FLEX for the screens and stuff. We have a multi-screen format so you can page backward to see what's scrolled by and a Hi-Res screen that will enable you to have a 2-line by 51-character display. That's better than an Apple!

We also implemented a full function keyboard, with a control key and escape key. All ASCII codes can now be generated from the Color Computer keyboard!

We also added some bells and whistles to Radio Shack's Disk system when you're running FLEX or OS-9. We are supporting single or double sided, single or double density, 35, 40 and 80 track drives.

MOVING FROM moves Color Basic from ROM to RAM. Because it's moved to RAM you can not only access it from FLEX, you can run it and even change it! You can load Color Computer cassette software and save it to FLEX disk. Single Drive Copy, Format and Setup commands plus an online help system are included.

Color FLEX includes an external terminal program that lets you use a standard terminal hooked to the RS-232 port. This will let you use a full sized keyboard with a 24x80 display. Your printer is then hooked to the terminal. The system will automatically control the printer. No hardware or software modifications are required.

Installing FLEX is simple. Insert the disk and type:

RUN "FLEX"

That's all there is to it! You are now up and running on the most popular disk operating system for the 6809. There are hundreds of software packages now running under the FLEX system. We have 100 packages ourselves. Open your Color Computer to a whole new world of software with FLEX.

**FLEX \$58.00
INCLUDES OVER 25 UTILITIES!
DOES NOT REQUIRE ADDITIONAL HARDWARE!
OPTIONS**

ED/ASM is a very powerful editor/assembler package. ED has all the features of TSC's editor with the addition of screen type editing, MACRO capability, and a math package. With the math package you can perform simple or complex formulas with the answer in HEX.

OSM

**OS-9/FLEX MACRO ASSEMBLER
by Frank Hogg**

For FLEX or OS-9. Create FLEX or OS-9 binary files from either FLEX or OS-9. OSM is a MACRO assembler like CRASMB. It is compatible with TSC's Assembler, but it has more powerful MACROS. OSM makes it easy to move FLEX programs to OS-9. In OS-9 it gives MACRO capability like TSC's assembler and is compatible with TSC source files. OSM was used by the author to move CRASMB to OS-9.

**PRICE \$125.00
Specify OS-9 or FLEX**

AUTOTASK

WITH MENU

**PRICE \$125.00
Includes source on disk
Manual \$10.00**

6502 TRANSLATOR
Translator 6502 code to 6809
\$75.00

SUPER SLEUTH
Disassembler for 6800/6809 or Z80
\$99.00

DECIMAL and BINARY? In its simplest form it can be used for base conversions. You can also create a MACRO and pass parameters to it. Works with files larger than memory. It has many additional features.

AMS is also compatible with TSC's assembler. It has MACROS and conditional asm. It has more powerful MACROS than TSC's. ASM was created by taking our CRASMB program and making a 6809 only version of it. Nothing else was removed. Both programs have been set up for FHL Color FLEX and cost ONLY 100.00.

DBASIC allows the use of the standard Disk Extended Color Basic under FLEX. All disk input and output operations are done through FLEX and are completely compatible with the normal FLEX utilities. This means that files and programs written to disk by DBASIC may be manipulated by FLEX editors, sort/merge, etc. It also means that these files are not compatible with standard Disk Color Basic files. However, the cassette files are compatible and provide a means of conversion. Also included is a DBASIC program to read a Radio Shack Disk and write to a FLEX disk.

All of the BASIC language components described in the Radio Shack manuals are implemented, with the following exceptions:

1. Random files are not supported. FIELD, LSET, RSET, etc. will be of no use.
2. BACKUP, COPY, and DISKIN are not implemented and will give syntax errors. Use the equivalent FLEX utilities instead.
3. A new BASIC command called FLEX has been implemented. FLEX will terminate DBASIC and return to FLEX.

a. DSKIN and DSKOS are completely implemented.

DBASIC is only \$30.00 when purchased with Color FLEX. \$40.00 later.

Other languages available include: FORTH, Pascal, Fortran77, C, A/BASIC compiler, plus more. Application packages include: A/R, G/L, A/P, Inventory, Electronic Spreadsheets, Accounting, Database programs and more. SEND FOR LIST.

TRS-80 COLOR COMPUTER COMPLETE WITH 64K RAM, 24K ROM, SINGLE DISK DRIVE AND FLEX, SET UP AND READY TO RUN FOR ONLY \$1,275. Includes 120 day extended warranty. If you have a Computer, call about RS disk controllers and drives.

TRS-80 COLOR COMPUTER

FORTH FOR THE TRS-80 COLOR COMPUTER DISK SYSTEM

Tired of getting control of your Color Computer? Tired of translating HEX to decimal? Tired of remembering where the VDG and SAM are and how to program them? Want to write machine language code with assembly language mnemonics instead of POKE's? Want to write programs in half the time? Want to write lots of small pieces of code that you can put together in seconds to do BIG JOBS? Want a language that is at least 5 to 10 times faster than BASIC? Want to learn everything there is to know about FORTH, with the best manual on the market, including lots of examples of FORTH applications, and detailed explanations of how everything works?

CC FORTH IS THE ANSWER!
Includes Editor, 6809 Assembler
String Functions, Disk Data File
Operations and Much Much More!

\$99.95

FORTH

FLEX COMPATIBLE FORTH

BY Chuck Eaker, Ph.D.

X-FORTH NOTES

Supplied on one 5" disk or 2 5" disks,
with a 400+ page manual.

Disk(s) have the source of everything but the core.
PRICE only \$149.95 plus \$2.50 S&H
Manual available separately for \$49.95 plus \$2.50 S&H

**We Have
DynaCalc
For Flex
\$200.00**

SOFTWARE CATALOG

PROGRAM	OBJECT/WITH ONLY/SOURCE	Code
BILLPAYER	169.95	x
PLOT	44.95	x
TABULA RASA	100.00	x
Mailing List	99.95	x
Forms Display	49.95	x
Inventory with Material Requisition Planning	100.00	x
Some Common BASIC Programs	69.95	x
Some Practical BASIC Programs	69.95	x
OSM OS-9/FLEX Macro Assembler	125.00	9
DBASIC (For Color FLEX only)	40.00	9
Intomag Data Base	295.00	x
Management System	295.00	x
Osborne Accounts Receivable	295.00	x
Osborne Accounts Payable	295.00	x
Osborne General Ledger	295.00	x
DynaCalc	200.00	9
UniFlex Simulator	110.00	9
FLEX For Color Computer X-FORTH (FLEX)	99.00	9
CC FORTH (TRS-80C for TOOLKIT #1 (BASIC))	149.95	889
TOOLKIT #2	99.95	9
AUTOTASK	49.95	69.95
A/BASIC Compiler	129.95	9
Extended Utilities	150.00	9
Password Protection	49.95	69.95
CRASMB (X Assembler)	69.95	89.95
Personality Modules (1 Inc)	139.95	9
6502, 6800, 6805, 6809, 280, 8080, 1802	50.00	9 88
CRASMB for OS-9	200.00	9
Personality Modules (6809 included)	35.00	35.00
6502, 6800, 6805, 280, 8080, 1802	100.00	9
ED/ASM Editor and assembler	100.00	9
HEADTAP	54.95	9
SPELLTEST	199.00	299.00
READTEST	54.95	74.95
ESTHER	39.95	59.95
HELP	29.95	49.95
Job Control Program	49.95	89.95
DYNASOFT PASCAL (FLEX)	59.95	89.95
DYNASOFT PASCAL (OS-9)	69.95	99.95
DYNASOFT Compiler Source	125.00	9
DYNASTAR Screen Editor (OS-9)	149.95	99.00
SUPER SLEUTH (6809/6809)	99.00	889
SUPER SLEUTH (Z80)	99.00	889
CROSS Assembler: Macros for TSC ASM B	49.95	each
6800/1, 6805, 6502, 280, 8080/5	99.95	9
3 FOR	75.00	9
6502 Translator	75.00	9 88
Debugging Simulators 6805 or 6502	75.00	9
STYLOGRAPH PH 2.0	295.00	9
STYLOGRAPH MAIL MERGE	125.00	9
STYLOGRAPH Spelling Checker	145.00	9

CODE X = XBASIC, 9 = 8809, 8 = 6800, P = PASCAL

Software by Technical Systems Consultants, Inc.	150.00
FlexTM (includes Editor & Assembler)	150.00
UniFlexTM (includes one year maintenance and update)	450.00
Editor	50.00
Assembler	50.00
6800 Cross Assembler on 6809	250.00
Text Processor	75.00
Extended Basic	100.00
Basic Precompiler (specify standard or extended)	50.00
Pascal (FlexTM)	200.00
Pascal (UniFlexTM) (Add \$75.00 for one year's maintenance and update)	225.00
SoftMerge Package	75.00
6809 FlexTM Utilities	75.00
Debug Package	75.00
Diagnostic Package	75.00
Software by Microwave Systems Corp.	
OS-9 TM Level One	200.00
Operating System	500.00
OS-9 TM Level Two	200.00
Operating System	125.00
BASIC9 TM	125.00
OS-9 TM Macro Text Editor	50.00
OS-9 TM Interactive Assembler	895.00
OS-9 TM Interactive Debugger	400.00
(Disk version)	
CIS Cobol Compiler	
Pascal Compiler	

USA add \$2.50 for standard UPS shipping & handling
Foreign orders add 20% Airmail
Specify 5" or 8" disk and 6800 or 6809
VISA - MASTER CHARGE - DINERS CLUB ACCEPTED

OUR SOFTWARE IS GIMIX COMPATIBLE

**FRANK HOGG
LABORATORY**
130 MIDTOWN PLAZA
SYRACUSE, NY 13210 (315) 474-7856

Flex User Notes

Ronald W. Anderson
3540 Sturbridge Court
Ann Arbor, MI 48105

MATH AGAIN

I promised that we would look at addition and subtraction this month, and I've done the routine to do both, since subtraction may be done with a complement and add procedure that involves only one extra step. Perhaps we had better review binary two's complement numbers just a bit. Let's consider 8 bit numbers for illustrative purposes. A signed 8 bit number may have a value between -128 and +127. You may remember that the negative of a number is obtained by complementing the number and adding 1. Complementing simply means changing all the 0's to 1's and vice versa. For example, 7 would be represented as:

00000111

complementing:

11111000

add 1

11111001

The result is the representation of -7. -7 is represented by 11111001. You will note that the leftmost bit of all negative numbers is a 1 and the leftmost bit of all positive numbers is a 0. That bit, is therefore called the sign bit. Now, the 68XX has an instruction called NEG for negate, which does the complement and add 1 operation. There is, however, one problem. There is no NEG0 operation, only a NEGA or NEGB (or NEG <memory location>). All these are single byte operations. In order to negate the contents of D, you must use a sequence of instructions.

```
COMB
COMA
ADD0 #1
```

In other words, the 16 bit instruction is missing. Of course, three instructions are not all that difficult to put together. To add two numbers input as sign and magnitude, we simply convert negative numbers to their two's complement form by the above three instructions, and add them. The whole operation is done in the program ADDSUB. There are several ways to add two 16 bit numbers. One of the sneakiest is to use a register offset instruction.

```
LOX ARG1
LDD ARG2 ARGS ALREADY CONVERTED TO TWO'S
COMPLEMENT
LEAX D,X
```

That is a very brief, and fast way to add two numbers. However, it is difficult to test for overflow, since LEAX doesn't change any but the zero flag. Assuming that the add routine is passed the two arguments on the User stack, a better routine would be, assuming the arguments are passed as two's complement numbers and the results are returned in the same way:

```
ADD LDD 0,U
ADD0 2,U
BYS OFLOW
LEAU 4,U REMOVE ARGS FROM STACK
PSHU D PUT RESULT ON STACK
RTS
ERROR LDX #ERMSG
JSR PSTRNG
JMP WARMS EXIT ON ERROR
```

The overflow flag indicates an overflow on signed arithmetic. The faster method works fine if you can guarantee that all arguments will be within the proper range so that overflow cannot occur, or if for some reason, it is understood that overflow is acceptable.

For subtraction, you use the old rule of Algebra, change the sign and add. That is, the number to be subtracted, which is in two's complement form, is negated and the two arguments are added. In the case of the above routine, we would simply insert the following above ADD.

```
SUB LDD 2,U
BSR NEGATE
STD 2,U
```

ADD etc.

```
NEGATE COMB
COMA
ADD0 #1
RTS.
```

The listing for ADSUB should tie all this together for you. Next time, we will go about putting together the multiply, add, and subtract routines as a group of subroutines, and run them with a small executive program that will get the arguments and operation and perform it.

LANGUAGES

I recently received a letter from Bob Nay at '68' Micro Journal, indicating that he thinks it would be a good idea for me to write a comparison or analysis of the various languages that are available for our SS-50 FLEX09 systems. I reminded Bob that I had done that about a year ago, but perhaps since the coming of the color computer, I should repeat that analysis from my present viewpoint a year or more later. I've "grown" as a programmer in that time, having taken part in the writing of the largest program I have ever been near, this past year and a half or so.

At least one reader of my last effort along this line, wrote to say that I had been unbiased in my presentation. I hope I can do as well this time. We ought to look at FORTH, BASIC, Pascal, "C", and Assembler, and compare the virtues of each. Of course, my personal preferences will show, but I will try to be objective. First of all, it ought to be said that the various high level languages were all designed for different purposes. I will mention the designer's purpose in the discussion of each language.

FORTH

FORTH was designed by an Astronomer in an effort to create a very compact language so that programming might be done quickly. Those objectives were certainly met. FORTH is a language all by itself. There's nothing quite like it. I've read with interest nearly everything others have had to say about it, and they have influenced my thinking somewhat. First of all, FORTH, like all the other high level languages, is system or computer independent. FORTH, in fact, is a little more uniform from one computer to another, because it also provides the operating system. The basic unit of memory in FORTH is a SCREEN. A SCREEN is a work area of 16 lines of 64 characters each. All numerical and logical operations in FORTH are done by means of a STACK. You put "arguments" on the stack, and then perform the operations. If you have ever used a Hewlett Packard calculator, you know about "Reverse Polish" notation. You don't enter 2 + 3 = to add those numbers, but instead, 2 ENTER 3 +. While you may think that to be backwards, isn't that how you do it with a pencil and paper? You write "2", then you write "3", then you draw a line, and add. Not really so different, is it?

Since my first Hewlett Packard calculator, I have been a fan of reverse polish notation. In FORTH, you would enter the variables "2 3 + ." The "." means "output the result" (equivalent to PRINT in BASIC). All arguments in a FORTH program are separated by spaces. You write a program in FORTH by defining words. "1" is a word that causes the addition of the two top numbers on the stack. Entering a number, places it on the stack. "+" pulls the two top numbers from the stack, adds them and places the result back there. You could define a word that multiplies a value by 2. You define a word by writing a "colon definition".

```
: 2TIMES 2 * . ;
```

The word just defined is "2TIMES", that is, the first argument in the definition is the name of the word being defined.

The definition is started with a colon and ended with a semicolon. Now if you had FORTH running and you entered "7 2TIMES" (return) you would get "14 OK". The "OK" means that FORTH has done what you asked and is waiting for another command. You defined a word that multiplies the top number on the stack by 2 and outputs it. By means of building new words, you can quickly build a rather complex program. Words must be defined before they may be used or referenced. In my opinion, FORTH programs are rather cryptic in appearance, and are difficult to read so that one programmer may not easily follow another programmer's program. Perhaps that is because a programmer really "invents" his own programming language as he goes. (see example below)

One of the reasons (for me) for not liking FORTH particularly, is that there are about 100 "words" in the basic vocabulary that you must "learn" before you can become proficient in FORTH. In my opinion, FORTH is rather "irregular". That is, related words do rather arbitrary things. (I have by now insured that I will receive 17 hate letters from FORTH fans.) It has been said that you either LOVE FORTH or you HATE it. I guess I am not in either category. I've grown proficient enough in it to be able to use it reasonably. I don't see the fantastic advantage of using it, over other high level languages that is claimed by the FORTH enthusiasts.

FORTH has a place in the list of programming languages. If you are doing some sort of process control, data collection, etc. and no one else has to read your code, you might find FORTH the most efficient language to use. It is almost as interactive as BASIC, in that you can change a word definition and "recompile" the program almost instantly so you can try it again. When you change a FORTH program, you actually edit the disk sector that contains the source. Recompiling it consists of not much more (by external appearances) than reloading the file.

FORTH proponents claim that programs may be written much more efficiently than they may be done in Assembler, in terms of programming time vs results. I think that can be said of all the high level languages.

Pascal

Pascal is a language designed to teach "structured programming" to computer science students. The very large program to which I referred above was written in Pascal, so I have more experience with it than with the other languages. Pascal is very precise, and models the "real world" rather well. You can define variable types that can have only certain values. For example, you can define a TYPE COLORS = (RED, BLUE, GREEN), and then a VARIABLE FLAG of type COLORS. Now if you should happen to attempt to assign the value 17 to FLAG, you will get an error, because you have defined FLAG to have the values RED, BLUE, or GREEN only.

I've thought about writing a book on Programming in general without reference to a specific language. I find that I always come back to Pascal for examples because Pascal statements declare precisely what they are doing.

Because Pascal statements are so clear and totally unambiguous, a program written in Pascal by one programmer, is quite easy to follow for another programmer. Pascal is a language that has a good standard, and most of the implementations have only minor variations. There are at least 5 or 6 implementations of Pascal available for our FLEX systems.

On the bad side, Pascal is very "wordy". If you are a slow typist, you might not appreciate it as well as some of the more "shorthand" languages. Also, standard Pascal has virtually no built in functions for handling strings. You must write your own. Though that sounds like a chore, I found it to be rather straightforward, and then found that I didn't very often need the string functions of BASIC, for example. I could build what I needed into the program rather simply. It has been said that Pascal is not very good for linking to hardware such as A/D converters, etc. The better implementations all allow linking to Assembler code, and therefore interfacing to hardware is very straightforward.

C

"C" is a relatively new language. It was written at Bell Labs. "C" Compilers for our FLEX operating system are just now becoming available. "C" is a more "concise" language than Pascal, but is not as easy to read. It is a language designed primarily for "Systems Programming", that is, computer operating systems and utility programs, such as disk copying, file transfers, etc. It of course may be applied to many other sorts of problems. This language doesn't have the extensive error checking of Pascal. Pascal will report array subscripts out of range, input numbers out of the required range, etc. C will not report an "overflow" of integer arithmetic, for example. That means that adding 1 to 32767 will result in a value of -32768 with no error reported! The programmer has to be more alert to such possibilities when using "C". Frankly, I have not had sufficient experience with a fully implemented and bug free "C" to know if I like it as well as Pascal or not. I strongly suspect, however, that I will like it at least as well as Pascal after some familiarization time. Some of the shorthand notation is shown below, comparing the Pascal fragment with the equivalent C fragment.

INDEX := INDEX + 1;	Index++;
INDEX := INDEX - 1;	Index--;
INDEX := INDEX + 7;	Index += 7;
BEGIN	{
END	}
INDEX : INTEGER;	int index;

Parameters are passed to subroutines (functions in C) in a manner very similar to that used in Pascal. C allows "pointer" variables. int *place; defines the variable place as a pointer variable to an integer. *place references the data in the location pointed at by the pointer. place references the address at which the data is located. If a variable is named "data", &data will get the address of that variable in memory. place = &data will assign that address to the pointer variable *place is now equivalent to data. All that is a bit confusing at first, but very powerful. Assembler code may be included directly in line within a C program.

Assembler

Assembler produces the fastest, most efficient programs but programming in it is very tedious. You can be an expert assembly language programmer in 8080, and know nothing about assembly language programming in the 6809. The language (instruction set) is intimately linked to the hardware of the processor. For this reason, Assembly language is not "portable". That is, a program written for one processor may not be run on another. Programming in Assembler involves a great amount of detail. One line of high level language may require half a

page of instructions in assembly language. Most microprocessor based products that are to be produced in any quantity, are programmed in Assembly language. The result is the highest performance with the least hardware.

BASIC

BASIC is the first language most people new to computers manage to learn. It was designed to be easy to learn for beginners. It has met its design requirements very well. BASIC is usually implemented as an interpreter. It has no compile step. Programs are simply written and run. The interpreter "interprets" the BASIC statements and generates the necessary machine code instructions to run the program, a line at a time. A compiler, on the other hand, analyzes the source code and produces a machine code "object file" which is then run. As a result, most compiled languages run or execute the same or equivalent program considerably faster than BASIC. BASIC is easy to learn and "interactive" because there is no compile step between the writing of the program and its running. I highly recommend that anyone who wants to learn to program a computer learn BASIC first. From the standpoint of learning to become a good programmer, it would perhaps be better to learn Pascal first, but most people want to be able to run their computer in a short time. BASIC is the shortest path to being able to do some fairly complex programs quickly.

Why use a compiled language rather than an interpreted one? Usually, the reason is speed of execution or reduced memory requirements. You don't need to have the whole interpreter in memory when you run the program. Some languages use an "in between" approach. They are neither compilers nor interpreters. Some implementations of Pascal are in this between category. The compiler generates what is called Pseudo code. That is a set of instructions for a hypothetical computer, that is theoretically the same for all the implementations. That set of instructions is then interpreted by a P-code interpreter when the program is run. Theoretically, the same P-code will run on any computer for which the P-code interpreter has been written. The results are intermediate in speed between an interpreted language and one in which the compiler produces "native code" (Assembler source code). To give you some idea of the relative speeds of some of the compilers, TSC 6 digit BASIC may be used as a standard. It is about the fastest BASIC around.

Lucidata Pascal, which is a P-code implementation, is about twice as fast as BASIC. OmegaSoft Pascal, and TSC Pascal, which are native code compilers, run the same program about 10 times as fast as that BASIC. "C" is in the general ballpark with the faster Pascal compilers, and FORTH is perhaps half as fast as the fastest of the pascals.

How efficient are the various languages in terms of size of program required to do the equivalent job? My rule of thumb is that 100 pages of assembler code roughly equal 15 pages of Pascal, 10 pages of "C", or about the equivalent of 5 pages of FORTH. I wrote my "famous" test program to find prime numbers in FORTH once, and it occupied 16 or 20 lines, as opposed to a page or more of Pascal.

How well do the compilers generate code compared to a good assembly language programmer? The better P-code compilers generate less bytes of code for a given program than the "native code" compilers. That is because they rely on the interpreter more heavily. That is, they use "pre written" subroutines extensively. My best estimate is that the better compilers generate three to five times the machine code for a large program, than would be generated by an assembly language programmer to do the same thing. Some of the compilers only load the subroutines that are called in the program, so that a small program that only uses integer

arithmetic, for example, may have a very small "runtime package". Others simply load the complete runtime package, so that the capability is there for floating point arithmetic, sines, cosines, etc. whether the program ever uses those functions or not.

Why use a compiler rather than BASIC? BASIC has several limitations, aside from being slow in execution. It has the limitation that variables may only have one or two letter names, with a single digit number appended. The very features that make BASIC easy to use, also make it less capable than other languages. Those two letter variable names are hard to remember and associate with their function. Most of the other languages allow long variable names. Compare the following statements.

```
IF HOURS > 40
  THEN PAY := 40 * RATE + (HOURS - 40) * RATE * 1.5
  ELSE PAY := HOURS * RATE;
```

(Pascal program)

```
10 IF H > 40 THEN 40
20 P = H * R
30 GOTO 50
40 P = 40 * R + (H-40) * R * 1.5
50
```

(BASIC PROGRAM)

```
If (hours > 40)
  pay = 40 * rate + (hours - 40) * rate * 1.5;
else
  pay = hours * rate;
```

(C program)

Aside from the use of upper case in Pascal, and lower case for "C", and a few minor syntax differences, the programs look pretty much identical. Most FORTH implementations have only integer arithmetic capability. While it is possible to write software to do money calculations by inserting a decimal point two places from the right, it takes quite a few word definitions to accomplish that. Given such a package, the FORTH equivalent would look something like this:

```
HOURS 40 > IF 40 RATE * HOURS 40 - RATE * 1.5 * PAY !
ELSE 40 RATE * PAY ! THEN ;
```

There are some BASIC compilers around. Many of these are quite "Pascal Like" in that they allow use of variable names and labels. My personal opinion, is that it takes considerable effort to become proficient in a language, and that that effort might best be expended on a "standard" language such as Pascal, C, or FORTH, rather than a unique "Extended BASIC" language. I find it hard to understand why someone would spend a great deal of time generating a unique BASIC compiler when with the same effort, a compiler for a "standard" language could be generated. There is one notable exception to this. If a compiler is written that uses exactly the same syntax as a good BASIC interpreter, the user has "the best of both worlds". He can write his program using the interpreter so that debug is quick. He has all the advantages of being able to change a line and run the program instantly, without any compile step. After the program is running correctly, it may be compiled for speed of execution and compactness!

There are a couple other languages available for our FLEX systems. FORTRAN is around somewhere, as is COBOL. FORTRAN is the original high level language, and as such lacks some of the niceties of the newer languages. It is quite usable, and if you are a FORTRAN programmer from way back, you may prefer it. I have had little exposure to, or experience with Fortran, and am not qualified to discuss it in much depth. PL/M or PL/C are available in some implementations. PL is rather

similar to Pascal. COBOL may be preferred by longtime business programmers. COBOL is VERY wordy compared to all of the other languages discussed here.

In quick summary, FORTH is rather cryptic looking. Pascal is very easy to read, but disgustingly hard to "short cut". It has so many built in safeguards, that it is very hard to "cheat". "C" is efficient, but lacks the safeguards that Pascal has. BASIC is easy to use, but it is hard to write a well documented clear program in it. On the others, I don't feel particularly qualified to comment.

I said something in my last attempt to do this sort of analysis and comparison, that bears repetition here. Fortunately, we don't all have to like the same thing. The only way to find out what suits you, is to try several languages and see what their advantages and disadvantages are. Depending on the type of programs you write, and your personality, you will like one of those languages above all others. If you like it, and it is easy for you to use, by all means, don't listen to what someone else thinks, just use it and like it.

COLOR User Notes

Robert L. Ney
5900 Cassandra Smith Rd.
Hixson, Tn. 37343

This month we'll look at several items for the Radio Shack TRS-80C Color Computer. We have a couple of new 'goodies' for the ROM Pak Slot; we'll look at some improvements on the Chips in the Computer from Motorola; and we have some New Product announcements, including a new Disk Operating System for this Computer from Star-Kits. We'll have to delay the Radio Shack Editor-Assembler report a little due to lack of time to adequately use it -- sorry.

I received a couple of items from one of our "Northern Neighbors", Ted Baleshta, 42 Herrington Court, Nepean, Ontario, Canada K2H 5N7. The first is little 2 Socket EPROM Board which he calls the K2468 Module. This is a simple little PC Board with two 24 Pin Sockets which can accommodate up to 16K of EPROM. It uses PC Land Cuts to select the type of EPROM being used. The Board is the width of slot in the Connector (a couple inches), and about 2 1/2" long. It is a neat little unit that has all the hard work already done. Contact him for pricing.

Ted also sent a Buffered Expansion Board for the ROM Pak Slot which he calls the Colour Buffer. This is a PC Board with Gold Plated Edge Connections to plug into the Cartridge Slot. It contains four IC's to provide full buffering for the Cartridge Lines, and has a standard 44 pin (22 on each side, .156" spacing) Female Connector on the output which will accept the standard 44 pin Proto Boards, such as the Radio Shack #276-1550. This will make Expansion of the Cartridge Interface easy and inexpensive. This is an item that has been needed for a while to make development a lot easier on the Color Computer. Again, contact Ted for pricing.

New Motorola Chip Designs, General Ramblings, etc.

Last month I mentioned that we had received a new MC6883 from Motorola - but I really didn't know what it was. Now I know. Back the first of the year I ran a couple of these Columns on the MC6883 and its capabilities, and discussed the Type #0 end Type #1 Memory Maps. If you remember, you can run the CPU Clock at 1.8MHz in the Type #0 Memory Map, either in the 'Address Dependent' mode, where part of the Memory Access Area runs at double speed, and part at normal speed, or you can run the whole 64K at double speed. The problem was that you could NOT run at 1.8MHz in the Memory Type #1, or ALL RAM Mode. WELL, NOW I CAN! This new MC6883 is supposed to run at 1.8MHz in The ALL RAM Mode. Since I just found this out about an hour ago, I

haven't had a chance to try it out, but at least Motorola is on the right track. The only thing really holding the Color Computer back is the CPU Speed; and this could be the development we need to really 'turn this Machine loose'. We can by-pass the Keyboard/Video Display restrictions with some Software (such as Star-Kits REMOTERM, to mention one of several) and a Terminal such as the ADDS series, TELEVIDEO series, ADM series, etc. The Color Computer is already capable of working with 96K of Memory; and the Memory Map switching capabilities I just mentioned are still UNPARALLELED IN THE INDUSTRY. Combine all of this with a 1.8MHz CPU Clock and 8" Double Density Floppy Disk Operation becomes "routine", along with Hard Disk Operation, Multi-User Capability, etc. Sound good?? It is easily possible, WITH A NORMAL CPU CLOCK RATE OF 1.8MHz!!

But, I would like to point out TWO THINGS. FIRST, Motorola needs to be convinced that the 1.8MHz ALL RAM capability is NEEDED (and that they need to put the MC6883 and its capabilities 'OUT ON THE OPEN MARKET'). This can only be accomplished by YOU, through requests for this, and other features, to Motorola. They are 'receptive' to new Ideas; the MC6809 has to be a prime example of this. LET THEM KNOW WHAT YOU WANT!! The 'man to contact' is:

Mr. Luis Bustamante
Product Marketing Engineer
Motorola Inc.
P.O. Box 20906
Phoenix, Ar. 85036

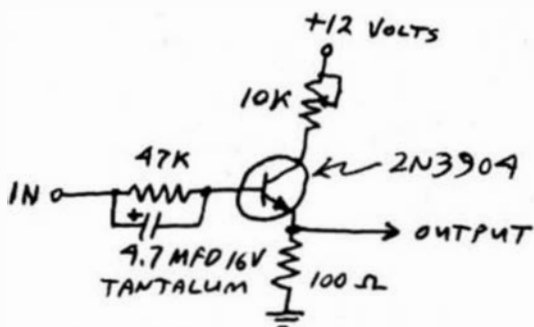
(Let me point out, also, that the MC6883 has an UNLIMITED POTENTIAL in the Control Applications field, too. We haven't really investigated the DMA Capabilities of the Chip, to mention just one of the other areas of possibilities. You "Systems Design" Types should give this Chip a HARD LOOK; it can solve a LOT of problems.)

The SECOND point I would like to make is this: while we have been 'examining possibilities' and 'developing' the Radio Shack TRS-80C Color Computer into a "Full Blown Computer System", I am NOT saying that this is the 'ultimate solution' to providing a Full Computer System for a Business Environment, either. If you have a VALID COMPUTER SYSTEMS REQUIREMENT, for goodness sakes, buy a Southwest Tech, GIMIX, Hellix, Smoke Signals, etc. It will be MUCH LESS EXPENSIVE and MUCH MORE POWERFUL than you will EVER be able to accomplish with the Color Computer. MY primary 'thrust' with this Column has been with two Ideas in mind:

1. That the Radio Shack TRS-80C Color Computer offers the most CAPABILITY per DOLLAR INVESTED of any Micro-Computer available Today, and that the manufacturer, namely Radio Shack, DID NOT KNOW WHAT THEY HAD when they came out with the Computer. (And I'm not throwing rocks their direction, either. They saw it as an Inexpensive Computer System for the ENORMOUS Home Computer market, and that is the 'target' they aimed at. I will also add that I feel they are doing a pretty good job of supporting the System WITHIN THEIR FRAMEWORK of WHAT THEY WANT TO ACCOMPLISH with the Color Computer.)
2. That the Radio Shack TRS-80C Color Computer offers a means for "John Q. Public" to begin with a Computer System for little more than the price of a 'Video Game System', and by adding a little here, and a little there, end up with a POWERFUL COMPUTER SYSTEM. Not only can he spread the cost over a long time period, but, HE CAN ALSO SPREAD THE LEARNING CURVE over a long time period. In essence, he can 'learn as he goes', while he 'pays as he goes'. AND, what he learns is DIRECTLY TRANSFERABLE to the previously mentioned FULL BUSINESS SYSTEMS. He is also learning about the most powerful 8-bit Computer Chip available today, and it will be around for many years. But, as I mentioned before, It will cost MUCH LESS MONEY to go ahead and purchase one of the FULL BLOWN BUSINESS SYSTEMS you see advertised in '68' Micro Journal if you ALREADY HAVE A VALID BUSINESS REQUIREMENT for a Computer System.

We have been hearing RUMORS (pretty solid ones, too) about a new MC6847 Video Display Generator Chip. I have not seen one yet, but understand that it has a 16 Color capability instead of the 8 we now have, and that it also contains greater Luminance controls, allowing much more flexibility in displaying shades, etc. I would appreciate any information anyone has on this subject, as it would add A LOT to the Graphics Capability of the Color Computer. We are already seeing the TRS-80C being selected over Apple, NEC, Atari, etc. for certain kinds of Graphics work. For instance, we have been working with an organization that needed an UNLIMITED number of Color Changes per line. It seems that most of the other Systems only allow a certain number of CHANGES PER LINE; they may offer more colors, but you can only change color so many times per scan.

Along these same lines, we have been running the Color Computer on the Amdex Color I and Sanyo Color Monitor the last month. These Monitors require a 1 Volt P-P NTSC analog input. Since the input into the RF Modulator is an NTSC analog voltage, the simple Emitter Follower driver shown below provides a simple modification that works well. I won't provide a full modification procedure (we'll include it in a later issue if there is enough requests for it); it is extremely simple and those accomplishing it will be able to 'stick it in' with no problem. Since the input is High Impedance, and output is Low, we have had NO loading or noise problems with the Mod. We install the Amp. just to the left of the RF Module in the Computer and bring a short lead out the Cassette Connector hole with an RCA Female Connector on it so that we can use the same RF Cable for the TV or Monitor, just plug it into whichever connector you want to use. The next step in this chain is to develop a method to get an R-G-B Output from the Color Computer; any of you with ideas here, send them in.



BWINDO

A "Smart" Disassembler for the Color Computer

Ron Levine Software
P.O. Box 356
Redwood City, Ca. 94064

Requires 16K w/ Extended BASIC
Tape: \$24.50

"BWINDO is a smart disassembler and cross-referencer with many unique features." This is the lead-in to the description of a Tape Program for the TRS-80C Color Computer that is 'different', to say the least. BWINDO is a Disassembler that is designed to do JUST ONE THING; help the user 'decipher' the BASIC and EXTENDED BASIC ROM's of the Color Computer. The output is not in a format that can be easily reassembled; it is in a format that makes INTERPRETATION easier. You don't have to spend hours and hours studying the disassembly to locate data areas, program areas, etc.; BWINDO already knows where these are. The data is displayed as 'Keywords', 'Jump Tables', etc. Floating Point constants are displayed both as they are stored internally, and in 'human-readable' form. Other data blocks contain their own annotations. BWINDO makes it easy to locate entry points of keyword routines, etc.

Another feature of BWINDO is that it provides TWO Cross-Referencing routines:

1. In the "storage reference" mode, you can enter ANY address in the 64K Memory area, and it will display a table of all ROM instructions which reference that address.
2. In the "transfer reference" mode, you can enter either an address, which provides a table of all the addresses that transfer control TO that address; or, enter two addresses and the table provides the addresses that transfer control ANYWHERE WITHIN the specified address range.

As I stated at the beginning, this Program is designed for ONE THING ONLY; STUDYING THE ROM's. When you load the Program, it begins execution by disassembling from \$A027 (the RESET Entry point). But, you have complete control of the Program; you can move back and forth with the 'arrow keys', 'Jump to any location to begin disassembly, etc. You can also either dump a display screen to the Printer or Initiate a continuous Printer output (which can be halted at any point by hitting any key). There are helpful notations added in the normal "comments" field of the output, pointing out 'Keywords', Control Characters, Floating Point Routine Info, etc. Personally, I feel the real 'strength' of BWINDO is the Cross-Reference capabilities. These allow you to find all references to any point in the system for studying what routines use what variables, which routines are called by which Commands, etc.

No, this is not a Disassembler that can be used for studying various programs, ROM Paks, etc. (The Micro Works has one that does an excellent job for that purpose). It IS an extremely powerful SPECIALIZED Disassembler designed for one purpose ONLY; studying the BASIC and EXTENDED BASIC ROM's in the Color Computer.

NEW Color Computer DOS Announced

Pete Stark, the Owner, Manager, Chief Programmer, Chief Admin. Officer, etc., of Star-Kits, P.O. Box 209, Mt. Kisco, N.Y. 10549, has announced the release of his "STAR-DOS"

for the Color Computer. (See the Bit Bucket this issue for his Product Announcement.) This should be an excellent product. We hope to have a full report in the next month or so.

Bob Rosen adds another system to his CONNECTION-80 Bulletin Board System

Many of you are familiar with the excellent CONNECTION-80 Bulletin Board that Bob has had on line for some time now (212-441-3755). He has added a second computer system to expand the capabilities; this is a 32K Color Computer with Three Disk Drives. The Software for the system was written by J. Blech, and will be available to anyone interested in setting up a similar system. You can access this system by dialing 212-441-3766, or if you have come in on the original -3755 number, and it is busy, you will be transferred automatically to the Color Computer System (if it's not busy). Bob also has some excellent products available through his SPECTRUM PROJECTS Co. to go along with all the myriads of info on the Bulletin Boards.

EIGEN SYSTEMS announces BASIC AID ROM Cart.

Again, look in the Bit Bucket for the Product Announcement from EIGEN SYSTEMS on their new ROM PAK. This Pak provides many features needed to efficiently program with the BASIC Language in the Color Computer, such as Automatic Line Renumbering, Single Key Entry of Commands, etc. One of the more powerful features is the MERGE Command, which allows the programmer to build 'modules' and merge them into programs as needed. This

makes 'Structured Programs' much easier; which makes debugging simpler, allows the use of 'proven' routines, and helps eliminate 'reinventing the wheel' each time you need a common routine. The \$35.95 price tag includes the ROM PAK, a Plastic Keyboard overlay, and a full Instruction Manual.

MARK DATA PRODUCTS releases another Game -- Astro-Blast

Yet another Product Announcement for this month (again, see the Bit Bucket), this one from MARK DATA PRODUCTS. Their Games have set 'Standards' for the industry, and this should be another good one.

COLOR CLINIC

TONY DISTEFANO
4680-18 Street
Laval West QUE
Canada H7R 2P9

It's two o'clock in the morning and you are typing away on your TRS-80 Color Computer. Your eyes are burning because you've been staring at that bright green screen trying to create your "Do everything program" for hours. So you turn down the color, contrast and brightness of the display but that doesn't help too much. It's still a big square of light. Well...what can you do? Follow these simple instructions and when you are finished you will have a reversed screen like mine.

Though these instructions are simple only those with soldering experience should attempt this project. You will need a Phillips screwdriver, a grounded soldering iron, solder, an IC extractor, two pieces of thin wire, flux cleaner, and a little patience. Oh by the way, opening your computer may void your warranty.

THEORY

Before you start tearing into your computer, a bit of background on the VDB (Video Display Generator) is in order. The VDB is a large scale integrated circuit (LSI) chip that takes care of all the video you see on the screen, be it Alphanumeric or full graphic. The VDB continually scans memory (Via the SAM) and displays what it sees. In the Alphanumeric mode it converts the ASCII code of a byte of memory into a graphic block that looks like the letter it represents. Normally an upper case letter or number is black with a green background. Lower case letters are the opposite, green with a black background. What my circuit modification does, is reverse the order so that upper case letters are green with black background and lower case letters are black with green background. This does not change anything in

memory nor does it interfere with Basic. It also does not change any graphic modes or color. Everything stays the same except the letters, numbers and symbols. The diagrams in this article pertain to any and all versions of the computer. Version 1.1, 1.0, 4k, 16k, 32k, 64k, BASIC, EXTENDED BASIC and even DISK BASIC are OK.

PART ONE. The Opening.

Before you start into this make sure that you have a large clean work space. Make sure the computer is not plugged in. Put the computer upside down on a soft surface. Unscrew the seven screws that holds the cover on. If you haven't opened it before, the seventh screw is under the black sticker that warns you not to open this thing. Turn the unit over again (top side up) and pick up all the screws that fall out. Put these aside in a safe place. Remove the top cover and put that away too. Lean forward slightly and gently pull up on the keyboard. Unplug the connector that ties the keyboard to the main board. Put the keyboard aside. Now cut the two tie wraps that hold the RF shield in place. That's the big square piece of metal with holes in it. Remove the RF shield and put it with the other parts. You are now ready for part two.

PART TWO. The Modification.

Before you start part two, let me tell you that the board is very sensitive to static electricity. Try to avoid dry areas and avoid touching the contacts on the board whenever possible. Ok, let's get going. Locate and pull out the 74LS02 IC marked U29 on the PC board. Carefully bend pins 1, 2 and 3 so they stand

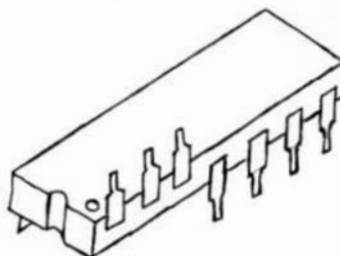


FIG-1

straight up in the air upside down. The dot denotes pin 01. If you are facing the computer it is the top left hand corner. See FIG 1. Now solder one end of a 4 inch piece of wire to pin 01 of the IC. The best wire to use is a #28 or #30 Wire Wrap wire. Solder another piece of 6 inch wire to pins 2 and 3. Yes both pins together. Now put the IC back in the socket. Make sure it is in the right orientation, the dot should be in the upper left hand corner. Also make sure that the wires and the pins do not touch the side of the RF shield. Now carefully remove the MC6847 IC marked U7 on the PC board. Bend pin #32 outwards just enough so that when you replace it, it does not go into the socket. Replace the MC6847. Again make sure of the orientation. The dot should be in the upper right hand corner. Take the other end of wire that connects to pin 1 of the 74LS02 and solder it to pin 32 of the MC6847. Take the other end of wire that connects to pins 2 and 3 of the 74LS02 and solder that to pin 2 on the MC6847. Be careful not to solder the pin to the socket. You won't be able to get the IC out if you do. Check the wiring and make sure that there are no shorts. Your wiring should look like the wiring in FIG 2. Now turn the power on. You should see the normal BIGN ON and copyright notice. Adjust the contrast, brightness and color on your TV so that you get crisp green letters with no background shading.

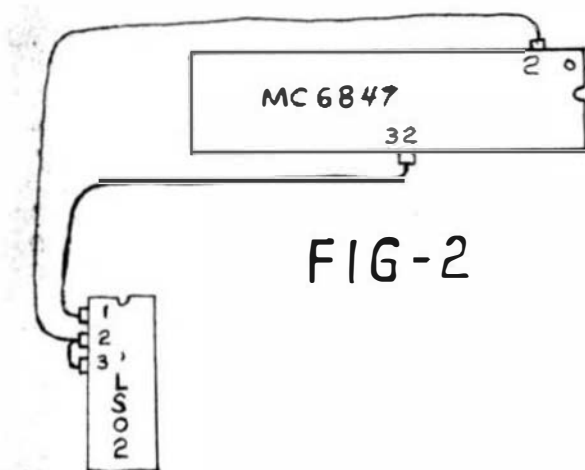


FIG-2

PART THREE. The Closing.

The closing up of the unit is the same as the opening but in reverse. Turn the power off and replace the RF Shield. Again, make sure that there are no wires hanging out and that there are no small pieces of wire or solder left in the closed area. Replace the keyboard and connector and put the lid back on. One thing to note, when you are putting the bottom screws on is that there are two short screws. They go under the keyboard. If you put the long ones there it will pierce a hole in the top cover. Do not overtighten them. After the computer is all back together again check all the functions just to make sure that all is running properly.

REPRISE

The whole operation should go off without a hitch, but if you do have problems drop me a line and I'll try to help you solve your them. Some of you might want to add a switch to be able to change back and forth between normal and reversed screen. To do so follow the wiring diagram in FIG 3. Make the wires long enough to be able to mount the switch on the outside cover. Warning!! Wires that run outside of the RF shield can cause interference with your TV.

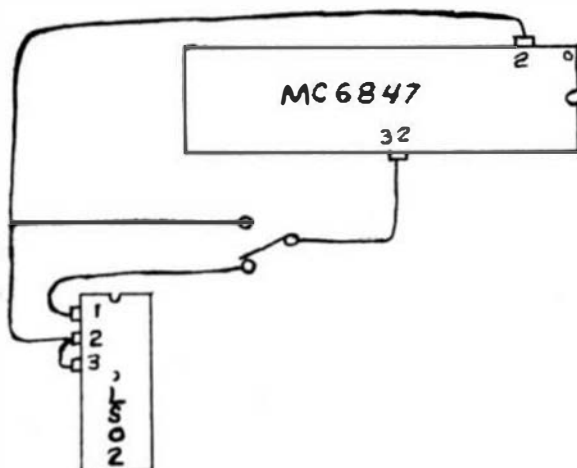


FIG-3

Coming soon! A small hardware project that will allow you to switch between normal and reversed screen in software. I.E. "POKE X,Y", where X = the memory mapped location and Y = 0 for normal screen and Y = 1 for reversed screen.

"C" User Notes

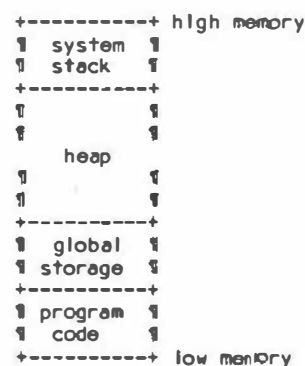
Norm Conno
3 Pryor Road
Natick, MA 01760

This month we will look at the runtime environment provided by some of the different compilers. Then an example of how this is used in a program.

What do I mean by the runtime environment? Primarily the initialization code before actually entering the user program at main(). This will usually involve setting up the stack pointer, initializing global and statics and other sorts of similar things.

Any stack oriented program, especially if it is also recursive, can chew up stack space. C is no different. You will find that most the runtime packages set the stack pointer to the location contained in MEMEND (FLEX location \$CC2B). This prevents running out of stack if there is empty space between user memory and \$C000.

Some also save the old SP value before setting it to MEMEND and set up a HEAP pointer that will can be used by the program for dynamic memory allocation. The heap is the space in memory between the user program (including variable storage) and the stack. This is shown below



Global (and statics) are treated a number of ways. Many of the Small C compilers generate code that reference globals as absolute addresses. The storage is generated with either RMB's, which are uninitialized; or with FCB's which set the variables to some predetermined value. Compilers that produce reentrant code, such as Introl, usually reference the global area via a pointer. Introl use the Y register and initializes the global pool to zero at runtime. Of course, if a compiler offers "Initializers" as defined in the C standard, then you have the option of setting the variables to whatever value that you want.

Many of the FLEX C runtime packages attempt to provide the user with some of the amenities of the UNIX operating system. These include command line parsing and I/O redirection. I have explained these briefly before, but let's review them in a little more depth.

I/O REDIRECTION

Both the Intral and Intersoft runtime packages support I/O redirection. The UNIX environment, as described in K&R(1), has three "standard" files associated with any program, which use the following descriptor

```
STDIN  -- the keyboard
STDOUT -- the terminal display
STDERR -- the terminal displays
```

Their values are defined in `STDIO.H`. These files are really I/O devices. Using these standard "files", you can test the program using your terminal. Once you are satisfied that it works, it may be used on real files by REDIRECTING the I/O.

Redirection is done from the keyboard when the program is called. Assume you have developed a program, named `caselo`, that translates all letters into lower case. It has been tested and found to work. Now you want to convert a file to lower case. The command line is

```
++caselo <infile >outfile
```

Within the program, all I/O is done via the standard terminal routines such as `getchar()` and `putchar()`. But the runtime package scanned the command line and found the redirection symbols. It opens `infile` and connects it to `getchar()`. Similarly, `outfile` is opened and connected to `putchar()`. Error handling in the event of problems opening any of the files varies with the vendor.

This is very similar to the `l` and `O` options of FLEX. The above command line could most likely have been accomplished with

```
++H Infile,O outfile,caselo
```

There is nothing wrong with this method, though I personally prefer the UNIX format. OS9, both levels one and two, support I/O redirection using the UNIX format.

Both the Intral and Intersoft terminal I/O functions look something like this

```
getchar()
{
    return(getc(STDIN));
}

putchar(c)
{
    char c;
    putchar(STDOUT,c);
}
```

In fact, if you want to, you can write all your terminal I/O directly using the file function `getc()` and `putc()` with the file descriptors `STDIN` and `STDOUT` respectively. Moreover, you could use `putc()` with the standard error file, `STDERR`. It is more verbose, but it has some advantages. Assume that you have a program whose output includes both data, prompts and error messages. Having the error messages and prompts imbedded in the data may be of little consequence if the output is being scanned at the terminal. But when you redirect the output, it may be quite undesirable to have it mixed. This can be avoided using the following technique.

All data is written using the functions `putc()` and `fprintf()` as follows

```
putc(STDOUT,a real char);
fprintf(STDOUT,"a real output string");
```

Error messages and prompts also use `putc()` and `fprintf()` but with the minor change

```
putc(STDERR,a real char);
fprintf(STDERR,"an error string");
```

At this point you can do just about anything you want with redirection. Consider the following line.

```
++program <infile >outfile >>errorfile
```

Here `infile` is connected to `STDIN`, `outfile` is connected with `STDOUT` and `errorfile` is connected with `STDERR`. FLEX has no equivalent for this.

One of the things about OS9 is that it does I/O redirection at the operating system level. Any program written for an OS9 system, in any language, may have the I/O redirected, it comes with the system.

COMMAND LINE PARSING

Redirection is handy, but the workhorse feature of these runtime packages is command line parsing. This makes getting at the FLEX command line arguments from within the C program a breeze.

The FLEX command line is parsed, with the arguments converted into valid C strings by terminating them with a `NULL`. The line may be stored in a buffer before the parsing, it depends on the particular runtime package. A count is kept on all the arguments and an array of pointers to each of the arguments is built up. Finally, the program is entered with the argument count and a pointer to the array on the stack. Consider the following FLEX command line

```
++program arg1 arg2 arg3
```

The argument count is 4 since the program name is included by convention. An array of pointer is built as follows

```
-----+-----
| 3 | pointer to arg3
-----+-----
| 2 | pointer to arg2
-----+-----
| 1 | pointer to arg1
-----+-----
| 0 | pointer to program name
-----+-----
```

Note that each pointer is two bytes. Suppose we had a program that started like this

```
main(argc, argv)
    int argc;    /* argument count */
    char *argv[]; /* pointer to an array */
```

`Main()` would be entered with `argc` set to four and with `argv` set to the address of `argv[0]`, which contains the pointer to the program name. Small C users beware, the declaration

```
char *argv[];
```

will probably get you into trouble. It states that `argv` is an array of pointers to `char`'s. Some Small C compilers can't seem to handle more complex definitions like this. You will have to use

```
int argv[];
```

since pointers and integers are both two byte quantities.

I have included two listings this month. Listing #1 is a quick cut at adding command line parsing to the Dug version #1 runtime package. It is by no means the most efficient way of doing it, but I wasn't acquainted with the quirks (if any) of FLEX's `NXTCHR` subroutine. In

does use NXTCHR, so I know it can be done. Also take note that it doesn't save the command line to a buffer. It changes the FLEX argument separators to NULL's within the command buffer. If you chain to command on the same line and the first used this code, it will blow up on you.

The second is a program called VU. This is a program for perusing through text file. It gives an some examples of how argc and argv are used. You might have to customize it a little for your system. I did not make use of gets() since I wanted the line to be terminated by a number of different command characters yet still retain some elementary line editing. I did this in C, but without using the BS and BE characters from TTYSET. The editing is designed for my particular terminal, an H19.

Users of Dugger version 1 will have to change the switch() to a series of "if then else" statements and the escape characters, "\n", "\r" and "\b" to decimal constants or equated symbols such as LF, CR and BS respectively. Other than that, and customizing the prompts for your terminal, it should not present a problem for the various compilers.

WHAT'S NEW

I am still waiting for Microwares OS9 C compiler. If you have read the some of the ads in 68 Micro I am sure you have notice that just about all the C compiler makers plan to have OS9 versions available at some time. I suspect that either Microware or Introl will probably be first to actually ship them, but I haven't check with the other vendors.

Some good news for stalwart 6800 users. I received a copy of a letter from 68 Micro reader Serge Stepanoff to Don. He has apparently developed a Small C compiler for the 6800 that generates pseudo code, which is then interpreted. More news will follow, or perhaps an article by Serge.

The next column will most likely cover some features of C not offered by Small C compilers. These are the "struct" and "union", as well as a few compiler operators such as sizeof().

NOTES

- (1) K&R stands for "The C Programming Language", by Kernighan and Ritchie, published by Prentice Hall.

LISTING #1

```

; C program entry code.
; It inits SP and UP, parses the
; command line into argc and argv,
; calls main(), and restores SP
; and UP on exit.
;
CINT JSR ZPCRLF
      PSHS U          save system UP
      STS FLXSTK,PCR
      LDS %CC2B       set SP to MEMEND
      LEAU -256,S      leave 256 bytes of stack
      CLRB            clr argv, and...
      TFR B,DP        set DP to a known page
      LEAX CCCAV,PCR   point to argv space
      LDY %FLXLIN      point to cmd line
CCIN1 LDA ,Y          end of line?
      BEQ CCIN3
      CMPA #13

```

```

      BEQ CCIN3
      BSR CCIN5       skip separators
      CMPA #13
      BEQ CCIN2
      LSLB
      STY B,X         stuff it into cccav[2B]
      LSRB
      INCB
      BSR CCIN7       find next separator
CCIN2 CMPA #13        end of line?
      BEQ CCIN3
      CLR ,Y+         no, set null and loop
      BRA CCIN1
CCIN3 CLRA           set null and call main()
      STA ,Y
      PSHU D          push argc
      PSHU X          push pointer argv array
      ASLB
      LEAX B,X        set end of array to null
      CLR ,X+
      CLR ,X+
      STX CCCEND,PCR  init free ram pointer
      LBSR MAIN

```

```

;
; exit()
;
; returns to FLEX after restoring
; the system stack pointers
;
EXIT EQU $
CCCEX LDS FLXSTK,PCR restore FLEX SP
      PULS U          and the USP
      JMP ZWARMS
;
; skip separators (space and comma)
;
CCIN4 LEAY 1,Y
      LDA ,Y
CCIN5 CMPA #13        skip blanks and commas
      BEQ CCIN4
      CMPA #',
      BEQ CCIN4
      RTS
;
; skip over a real argument to find
; the next separator
;
CCIN6 LEAY 1,Y
      LDA ,Y
CCIN7 CMPA #13        find next blank or comma
      BEQ CCIN8
      CMPA #13
      BEQ CCIN8
      CMPA #',
      BNE CCIN6
CCIN8 RTS

```

```

FLXSTK  RMB  2      flex SP saved here
CCCCND  RMB  2      pointer to free ram
CCCAV   EQU  1

```

VU command	Action
<cr>	list another line of the file
<space>	list another window of the file
^D	list 12 more lines

VU.CMD is a file perusal program. With it you may glance through files casually without needing to be 'quick on the finger' with the pause key! One of the biggest limitations of VU is that it is strictly one way. You cannot backup because only one sector at a time is buffered.

VU moves a "window" through the file. Initially the window is set to 23 lines. After each command, VU prompts the user with "--more--" if there is more of the file to be viewed, otherwise it returns to FLEX. Some commands may take an optional number in front of them. No <cr> is necessary after a command key.

VU command	Action
<cr>	list another line of the file
<space>	list another window of the file
^D	list 12 more lines
nn<space>	list nn-1 lines of the file
nnF	skip nn windows of the file
nnS	skip nn lines of the file
nnZ	change the window to nn lines
Q	quit to FLEX

```

/*
 * vu.c      rev: 1
 * a f commo
 *
 * created:   Nov-7-81
 * last edit: Jul-2-82
 *
 * A program to aid perusing disk files.
 *
 * 0/ 2define CRTSIZE 24 2define FOREVER while(1) 2define BOOL int 2define
 * METACHAR int 2define TRUE 1 2define FALSE 0 2define ERROR -1 2define EOF -1
 * 2define FILE char

```

```

FILE fcb;
main(argc,argv)
int argc;
char *argv[];
{
    int i;

    if (argc < 2)
    {
        printf("Usage: vu arg [args]*");
        exit();
    }

    i = 0;
    while(++i < argc)
    {
        if ((fcb = fopen(argv[i],"r")) == ERROR)
            printf("Error opening file: %s",argv[i]);
        else
        {
            display();
            fclose(fcb);
        }
    }
}

```

'68' Micro Journal

```

display()
{
    int linecnt, maxcnt, newmax;
    char c, anstg[10];

    maxcnt = newmax = CRTSIZE;
    linecnt = 1;
    FOREVER
    {
        while (linecnt++ < maxcnt)
        {
            if (dumpline(TRUE) == FALSE)
                return;
        }

        prompt();
        maxcnt = newmax;
        linecnt = 1;
        c = answer(anstg);
        if (isdigit(anstg[0]))
            maxcnt = atoi(anstg);
        switch(c)
        {
            case ' ' : break;
            case '^' : killprompt();
                        printf("skipping 2d screenfulls",maxcnt);
                        maxcnt = maxcnt & newmax;
                        skiplines(maxcnt);
                        maxcnt = newmax;
                        break;
            case 'q' : killprompt();
                        return;
            case 's' : killprompt();
                        printf("skipping 2d lines",maxcnt);
                        skiplines(maxcnt);
                        maxcnt = newmax;
                        break;
            case 'z' : newmax = maxcnt;
                        break;
            case 'r' : linecnt = maxcnt - 1;
                        break;
            case '^004' : maxcnt = 11;
                           break;
            default : printf("^007");
                       break;
        }
        killprompt();
    }
}

skiplines(lines)
int lines;
{
    while (--lines)
    {
        if (dumpline(FALSE) == FALSE)
            return;
    }
}

dumpline(mode)
BOOL mode;
{

```

```

METACHAR c;

FOREVER
{
    if ((c =getc(fcb)) == EOF)
        return(FALSE);
    if (pmode == TRUE)
    {
        putchar(c);
        if (c == 'r')
            putchar('\n');
    }
    if (c == 'n')
        return(TRUE);
}

}

answer(s)
char *s;
{
    char c, *p;

    p = s;
    FOREVER
    {
        c = getchar();
        if (isdigit(c))
        {
            *p++ = c;
            continue;
        }
        else if (c == 'b')
        {
            if (p > s)
            {
                printf(" b");
                p--;
            }
            else
                printf(" ");
            continue;
        }
        else
        {
            *p = '0';
            return(tolower(c));
        }
    }
}

/*
 * the following two routines must be
 * change for the particular terminal
 * in use.
 *
 *
 *
 * 1) set inverse video
 * 2) write the prompt
 * 3) restore normal video

```

```

*/
prompt()
{
    printf("#33p--more--#33q ");
}

/*
 * 1) set cursor at beginning of line
 * 2) erase the line
 */
killprompt()
{
    printf("#33l");
}

```

TELECON C

Paul M. Urle M.D., Ph.D.
 216 East 140 North
 Lindon, Utah 84062
 801-785-6067

The C programming language developed at Bell Laboratories was originally designed for, and implemented on, the UNIX operating system on the DEC PDP-11. C is becoming more popular on microprocessors especially with the appearance on the market of several reasonably priced compilers. These compilers have been designed to operate on 6809 FLEX and OS-9 systems as well as other processors such as the 8080, Z80, and 8086 under CP/M. However most of these compilers have implemented only a subset of the language as described by Brian W. Kernighan and Dennis M. Ritchie in the book *The C Programming Language*. The Telecon System's C compiler is available for 6809 FLEX, 8080, 8085, Z80, PDP-11, and soon on 6809 OS-9. This compiler is one of the most complete implementations available on the market and lacks only bit fields, multidimensional arrays, floating point and double precision floating point. Telecon says that floating point and multi dimensional arrays will be implemented by July 1982.

The input to the compiler is standard C source, compatible with UNIX, version 7. The output is an assembly language file in which the C source may be intermixed as comments which is then assembled to produce an executable file. The compiler requires approximately 48K bytes of ram, which includes all stack and symbol table areas. The following data types are available: char (8 bits), short int (16 bits), long int (32 bits), unsigned (16 or 32 bits), array, pointer, structure, and union. The floating point data type will be 32 bits with a 7 bit excess 64 exponent and a 24 bit mantissa. Constants may be expressed in the form of strings, characters, decimal, octal, or hexadecimal numbers. The entire set of unary, binary, and assignment operators are implemented including sizeof expression and sizeof (type-name) or casts. The available storage classes include: auto, static, extern, register, and typedef. Static and extern variables may be initialized at compile time. A complete set of program statements are implemented, including: if, if else, while, do while, for, switch case, goto, default, break, continue, and return.

The preprocessor is extensive, including #define with replacement string and replacement macro, and undef. Conditional compilation expressions with #if, #ifdef, #ifndef, #else, and #endif are

available. There are two other constructs, `#text` and `#endtext`, which allow non-C text to appear in the source program, such as assembly language code, and is similar to the `#asm` and `#endasm` implemented in other compilers. The I/O library is rich in functions, with both character puts and gets, and file formatted input and output. In memory formatting is also available. Numerous string and character handling functions are implemented in the library.

The compiler is supplied on both 5" and 8" standard FLEX disks. The compiler occupies 146 single density sectors. The standard I/O library is found in source form and is available from one of two files "stdio.h" or "stdio.l" (33 and 27 sectors respectively). The "stdio.l" file has less features than the "stdio.h" file and only one is included in a C source program. The runtime library and FLEX Interface ("rlib09.s" and "fix0el.s") are in source form and require 22 and 18 sectors respectively. When the compiler is invoked a "shell" is entered which provides a UNIX compatible environment for C programs and allows UNIX command lines with command line parsing and I/O re-direction. When the "shell" is entered a ++ prompt is displayed on the console. For example, to compile a C program to compute the prime numbers between 3 and 16381 using the Sieve of Eratosthenes algorithm found in BYTE, September 1981, the following steps would be required.

```
+++cc30fx (Invoke the compiler)

++ sieve.c rlib09.s fix0el.s -a -c >sieve.s (Input
command line)

+++asmb,sieve.s,sieve.cmd,+LS (assemble the 6809
source)

++[cr] (carriage return starts program execution)
```

10 iterations

1899 primes

+++

The command line in the "shell" provides switches (-a, -c, -l, -n) which permit absolute or relocatable output and other features. Following assembly, the 6809 source file (sieve.s in this example) which occupies 98 sectors may be deleted leaving only the executable file (sieve.cmd) which occupies 15 sectors.

A comparison among this compiler, Duggers C version 2.1, previously reported 6809 Pascal compilers (from 68 Micro Journal, Nov. 1981) running at 1 Mhz and Z80 compilers running at 4 Mhz (from Byte Sept. 1981) is shown in Figure 1. The Telecon C compiler is the fastest for the 6809 and does relatively well against the Z80 compilers.

Another benchmark for computing random numbers using a linear congruent sequence is given in Figure 2. This program produces a sequence of 126 random integers and test the speed of the single

precision (16 bits) Integer arithmetic. The results of this benchmark run on the Telecon C and Duggers C are shown in Figure 3.

The compiler displays diagnostics only on the console and permits only one error per line, which is a definite advantage since one syntax error may cause multiple diagnostic lines to be generated in some other compilers.

The manual supplied by Telecon is brief but complete enough to allow one to use the compiler. It contains a short description of the C programming language features implemented in the compiler. There is not a list of diagnostic messages in the manual and only a brief description of the UNIX "shell" command line syntax. The manual does provide a description of parameter passing and the stack format which permits external routines to be easily constructed and interfaced. The features described in the manual appear to be implemented in the compiler and to operate correctly.

The compiler is currently available from TELECON SYSTEMS, 90 East Gish Road, Suite 25, San Jose, California, 95112. The full version costs \$350.00 and an "Integer only" version (which cannot be upgraded to the full version) costs \$200.00. The compiler prices include free updates for one year.

This compiler makes available a very complete version of the C programming language to the 6809 user at a reasonable cost. The code produced is fast and relatively compact. I would definitely recommend this compiler for users who require the full features of the C programming language.

References 1. Kernighan, B. W. and Ritchie, D. W. The C Programming Language Prentice-Hall, New Jersey, 1978
2. Glibreath, J. A. A High Level Benchmark. Byte 6(9): 180-198 Sept. 1981
3. Anderson, R. W. Flex User Notes. 68 Micro Journal 3(9): 9-11 Nov. 1981
4. Elbert, T. F. Simulation, Games, and Random Variables 68 Micro Journal 3(9): 20-22, Nov. 1981

Figure 1 Prime Number Benchmark Comparison

Compiler	Execution Time	Total Bytes
Digital Research PL/1	14.0 sec	5977
Whitesmiths C	15.5 sec	7384
BO Systems C V1.32	49.5 sec	3932
Telecon C	43.4 sec	13484
Duggers C V2.1	65.4 sec	21784
TSC Pascal	59.0 sec	14334
Omegasoft Pascal	66.0 sec	2465
Dynasoft Pascal	143.0 sec	1490
Lucidata Pascal	158.0 sec	3929

8 4Mhz Z80.

Figure 2 Random Number Benchmark Program

```
#include (<stdio.h>)
int rintf(126,seed,an,cn,mod,niter;
main()
{ int i,xn,iter,k,count;
  seed=100;
  an=3;
```

```

cn=0;
mod=127;
niter=10;
printf("%d iterations\n",niter);
for(iter=1;iter<=niter;iter++)
{
  xn=seed;
  for(i=0;i<=125;i++)
  {
    xn=(an*xn+cn)%mod;
    rint[i]=xn;
    count++;
    for(k=0;k<=i;k++)
    {
      if(rint[i]!=rint[k]) count++;
    }
  }
}
printf("%d random integers",count);

```

Figure 3 Random Number Benchmark Comparison

Compiler	Execution Time	Total Bytes
Duggers C V2.1	24.6 sec	13864
Telecon C	31.2 sec	5613

Editor's Note: Since this was set we were informed that Telecon systems has decided not to support OS9 at this time. Interested users in an OS9 version should contact Telecon direct.

DMW - - -

SIMPLE WINCHESTER INTERFACE

The following software & hardware can be used as a guide to connect a Winchester hard disk drive to a 6809 system. The system consists of a Percom SBC-9 MPU, a Boaz 64k ram, a Western Digital WD1000 controller & a Shugart SA1002 5 meg drive. (The last two items are available from computer Dynamics for about \$1100.) The entire system- cpu, drive, controller, 64k ram, back plane & power supplies cost less than \$1650! The Winchester controller connects directly to the Percom's buffered IO. A description of the software is:

HARDFORM- Formats the disk, links, checks for errors & sets up the system information sector & directory.

FDRIVERS- The disk driver & terminal I/O.

PUTLDR- Writes whatever (the loader) is at \$C100-\$C1FF to track 0, sector 1.

HLOADER- Append this to TSC's loader, assemble, GET it and use PUTLDR to put it at track 0, sector one.

HBOOT- Boots FLEX from side 0.

CFAST- A program written in C which prompts the user for an interleave number, checks the disk for CRC errors and reports them. (useful for floppies too.) This program is not necessary to get the system running.

Selecting drive 0 selects side 0 of the SA1002 & access to drive 1 selects side 1 of the Shugart. IE:

TYPE	DRIVE	SIDE	: FLEX DRIVE
WINCHESTER	0	0	: 0
WINCHESTER	0	1	: 1
FLOPPY	1	0	: 2
NOT USED IN THIS VERSION			: 3

The requirements of the power supplies for the controller are:

+5 volts-

3 amps for the controller & 3 amps for the each 5 meg drive used. The current varies by about two amps while stepping, so this supply should have short lengths of 14 wire.

+24 volts-

3 amps max. Shugart says 24 volts +/- 3 volts. Although this drive may work at 21 volts, it will not tolerate more than 100mv of ripple. This supply must be very well regulated as the very high speed stepping rates cause high peak currents.

-8 to -15 volts-

The drive has an on board regulator, so get this from the host along with the buss.

Headaches:

The first problem occurred when I attempted to do a high speed restore. Normally during a seek the drive will accelerate & decelerate to the desired track. During a restore the drive does not know in advance where the destination is and cannot decelerate. During a restore a slower stepping rate must be used to prevent the drive from crashing past the opto-detector. The drive does a power-up restore & the controller does it's own error recovery, so one seldom needs to restore the drive.

A sloppy 24 volt supply caused a few seek errors & a lot of head scratching.

The only other problem was with the drive itself. There was a bit of solder mask on the edge connector foil which caused some problems until I scraped it off.

Because this system has never made an error I do not verify any writes, but I do a disk test every few days. (CFAST will test the disk in 35 seconds) A write verify requires a complete disk revolution before another sector can be written, resulting in a very slow write time.

The above software is available on 5" or 8" disk for \$15. An SS-30 interface card (unpopulated) is available for \$25

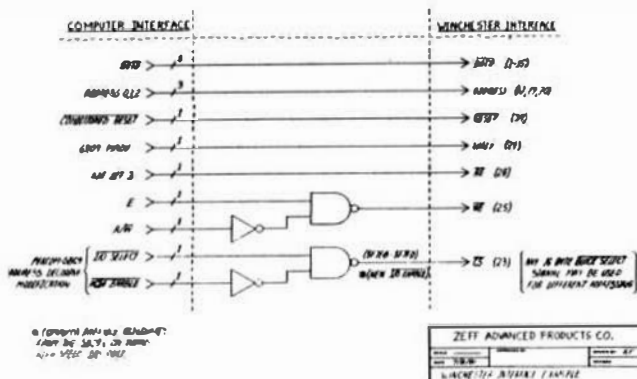
DIRECTORY OF DRIVE NUMBER 3

DISK: WINCHEST #1 CREATED: 19-JUL-82

FILE#	NAME	TYPE	BEGIN	END	SIZE	DATE
1	README	.TXT	01-01	01-02	2	19-JUL-82
2	COVER	.TXT	12-06	13-08	13	20-JUL-82
3	PUTLDR	.TXT	08-06	08-09	4	19-JUL-82
4	HARDFORM	.TXT	05-08	08-05	28	19-JUL-82
5	HLOADLIB	.TXT	08-0A	09-04	5	19-JUL-82
6	HBOOT	.TXT	09-05	09-09	5	19-JUL-82
7	CFAST	.TXT	09-0A	0E-09	50	19-JUL-82
8	CFAST	.CMD	0E-0A	10-09	20	19-JUL-82
9	CFASTTST	.TXT	10-0A	12-05	16	19-JUL-82

FILES=9, SECTORS=143, LARGEST=50, FREE=197

Robert Zeff
Zeff Advanced Products Corp.
2135 Stone Ave.
Modesto, Ca. 95351
(209) 577-4268



Editor's Note: This was submitted by Bob Zeff as a follow up to the advertising of Computer DYNAMICS, Greer, S.C., last month. Also see their advertising this issue.

The interface drawing and a copy of the directory of the disk with the software, as received, are shown here. Due to space requirements, it is impossible to run the complete source programs as an article. I therefore recommend that if you are interested in this project, then you might consider ordering the information (source and binary) direct from Bob Zeff, address as above.

DMW - - -

SPEAK & SING

Better than a year ago we received, for review, the Speech Systems Speak 'N' Sing 2 synthesizer board for the S30 bus. Due to a possible mix-up, on our part, the review of this product is just now being published. For this we apologize!

We have tested this board in both 1 and 2 mhz systems, both 4 and 16 byte address configurations and it and all the supplied software (which there is a bountiful supply) worked without error.

The heart of the system and the most expensive is the SC-01 Speech Synthesizer (about half the cost of the board) which sells for \$219.95 assembled and tested with manual and diskette containing all the necessary software to program and operate the board, either for speech or music. Two additional music and sound effect diskettes are available with single and 4 voice music selections as well as some sound effect programs.

The board as received is addressed (software) to function in slot #2 of the S30 bus position. If the board is placed in any other slot there are two ways in which it can be accessed. First, each time you call a binary program it prompts for the I/O board slot, this is to be answered with the port #. Second, most all programs have included the source file. This allows the user to change the port address in source and then assemble the binary file. Of course programs running in TSC Extended BASIC need only the port assignment changed in the BASIC source. We found that all of the BASIC programs furnished use line #120 as the port assignment line. Most binary programs have the port address at either \$C11D or \$02B1.

BASIC

120 PO=HEX("E020"): REM PORT ADDRESS

change to

120 PO=HEX("E0XX"): REM "XX" is slot address

MANUAL

The users manual furnished is very complete. It starts out by explaining some hardware considerations that may be necessary on some of the older Standard S50 Bus computers. In fact we know of one user who after installing the simple capacitor fixes shown in the manual discovered that a lot of previous "glitches" went away.

The manual consists of nearly 60 pages of detailed instructions on programming the board. This brings us to an important program furnished that should be explained at this point.

SSEEDITOR

The SSEEDITOR is a special program (furnished on the supplied diskette) that allows fast and efficient programming of the board. It is referred to as a 'Phoneme' editor. An 800 word dictionary is also supplied. This dictionary allows the immediate development of words, sentences and other speech functions and is accessed automatically by the editor. The dictionary being a standard ".TXT" file may be amended as necessary by the user. When entering the editor the port address of the board may be indicated by answering the "PORT?" prompt by either the port number or the hex address of the board assigned port. Having answered the port query the editor comes up in the "HELP" page mode, this mode can be recalled as required indicates the following commands

'up arrow' ↑	Enter Command Mode
+ CMD	Exec FLEX command
A	Enter Append Mode
C	Clear Buffer, enter Append
Mode	
D #	Delete Phoneme #XXXXX
E	Exit to FLEX
G NAME	Get Disk File
H	HELP Listing
I # PH	Insert Phoneme(s) behind #
K	Key Words Display
L	List Phoneme Buffer
O MSG	Output Message to Printer
P	Print Phoneme Buffer on
Printer	
R	Remove Inflections from Buffer
S NAME	Save File to Disk
T (1,2,3,4)	Talk Out Buffer
V	Vocabulary DEMO (esc):

The "up arrow" serves as the prompt in the command mode, where most of the programming is done. The commands available are then called as shown above. For example if the "V" command is called then the system speaks all the words (800) in the dictionary. This alone requires about 7 minutes of listening. The "ESC" key will abort this at any time. The editor establishes a buffer on the disk assigned as the work disk. Here is where all the words that are either used from the dictionary or written by the user are held for testing, editing, etc. Example; If we should enter the APPEND mode by typing "A" then we could append, to the end, whatever is in the buffer.

'THIS
'IS
'A
'TEST

Would be appended in the buffer. The " " is inserted before each word to tell the editor that the

text is a word and not phoneme code. If the words (any or all) are in the editor they would be used in their phoneme coding but displayed in word form. If any of the words are NOT in the dictionary then the user is informed "NOT IN DICTIONARY". At this point the user will code the word in, in phoneme code. To exit the APPEND mode type in (up arrow)T and the buffer will be output as speech. One other important aspect of this type programming is that the words are not automatically entered into the buffer with spaces between words or sentences. This is accomplished as:

```
'THIS*
'IS*
'A*
'TEST*
```

The "*" indicates that spacing, normally about 47 milliseconds in time, is to be inserted between each word. Also the following could have been typed in:

```
'THIS IS A TEST.
```

It would be reproduced as the example above. One other nice function is that the "." is noticed by the program and a 185 millisecond pause is inserted, knowing that the pause between sentences is longer than between words.

Additionally the actual phoneme could have been typed in, of course necessary if the word is not in the dictionary or you desire a different inflection to the word; i.e., Texan, Southern, Yankee, etc. The normal dictionary speaks 'mid-western American', whatever that is ya-all. An example is the word AMERICA:

```
UH1 M EH1 R I2 K UH1
(AMERICA - Phoneme)
```

To add some thing beside 'mid-western American' we would add inflection levels (1-4). they would be entered:

```
2/UH1 2/M 3/EH1 2/R 2/I2 1/K 1/UH1
```

The slash character is optional, but it helps readability.

The "D" delete command allows a range specified such as 20- 45, meaning that all words (actual or phoneme) between the 20th to the 45th are to be deleted. The "K" command causes the "Keyword Index" page to be displayed. Example; the phoneme mnemonic "B" sounds like the "B" in BAT or RUB. Believe me this is a real help and makes typing code a lot easier. The index is fairly extensive.

All the other commands are fairly self explaining with the exception of the "T" TALK command. The "T" command allows the buffer to be spoken by the system. Also the user may designate a certain point, in the buffer, at which the system should start speaking. Example; T26, means to start speaking with the 26th word in the buffer. Should the user type T2 26 then the words starting at the 26th word would be spoken but, the "2" scale inflection would be used. Inflections that have other coded inflections will still be spoken at the "2" scale. The inflections are programmed so that the higher the inflection number (1-4) the lower the inflection.

SPEECH SYNTHESIS

Needless to say that we are not going to cover the entire subject of "speech synthesis" as it would require far more pages than are in this issue. However, I will touch on a few important aspects of this subject.

There are several ways in which speech may be programmed into a computer. The "waveform" method was one of the first used. The voice waveform is

constructed by actually recording the analog speech and then processing it by A/D methods and storing it in memory. This system is by far the best quality and is in fact used in digital recordings which are becoming the rave among music buffs. The drawback to this is that the method is very expensive in the utilization of memory and despite many different compression methods devised it still uses far more memory than the other two major schemes.

Another method is referred to as "linear predictive coding" and is used by the TI "Speak and Spell" toy. The principle nature of this method takes advantage of the fact that human speech is very redundant. There are a few major waveforms for human speech, linear predictive coding removes this redundancy and uses only the critical human waveform data. The primary disadvantage is that this method requires hardware that limits the vocabulary, as each word (as the system explained above) must be actually prerecorded and then processed by an A/D - D/A system. Most predictive systems are coded into ROM and contain many standard words. However, the cost is very high, per word, for the coding of special ROMs, something on the order of several hundred dollars or more per word, for custom programming. This renders predictive coding schemes too expensive for the average micro-computer user.

This brings us to the third method, which is the scheme used by this board. This is by the "Phoneme" method. The phoneme being an isolated speech sound requires very little data to produce intelligent speech. Also the vocabulary becomes almost limitless and is, for all practical purposes, limited only by the desires of the user and digital storage capacity available on the host computer system. Cost wise this is the most effective method of generating speech. The heart of this board is the SC-01 speech synthesizer chip.

A short but useful and interesting (if you are into speech synthesizing) section is included in the manual. This section goes into enough detail concerning the different aspects of "phoneme programming" to give the user a feel for phoneme coding.

Basically the board is capable of producing 64 different phonemes. These in turn are divided into 6 categories:

```
VOICED
VOICED FRICTIVE
VOICED STOP
FRICTIVE STOP
FRICTIVE
NASAL
```

Example; VOICED phonemes are spoken using mainly the vibrations of the vocal chords. Phonemes that are produced by an articulator, such as the teeth, are known as FRICTIVE. The phoneme "F" being an example. A special program is included "PHNGROUP" that speaks the different phoneme groups. Words are developed by stringing phonemes together. The beginner will soon find that by listing out the words in phoneme code and by listening to the programs provided, he/she will soon be coding phonemes quite proficiently. I do not mean to imply that it is "Idiot simple" but anyone who can program in assembler, BASIC or any other language can soon catch onto phoneme coding. If I can, anyone can!

A normal coding session consists of first determining if the desired words are in the dictionary (actually you will be informed if they are not). Say out loud each word as it is being coded. Repeat as necessary until you begin to "feel" what the word sounds like. This requires a little getting used to at first, also other people (as well as most dogs and cats) will, at first, think you have suddenly become a stroke victim or something, vainly trying to communicate, however, a short explanation on your part (in plain, not phoneme, language) should clear

the air, more so with other humans rather than pets. My dogs still sit up and give me that special look each time I get into a coding session, fact is I honestly believe that they now understand some words in phoneme sound better than in my normal English(?).

Once you have what you feel is the proper code for the word then you can have the editor speak it back, just in case. As strings of words are developed and played back for approval it will be discovered that things are "really" getting easier. Because of the extensive possibilities allowed by the programs furnished, I will not attempt to cover coding any deeper, however, it should be pointed out that coding in phonemes is actually much simpler than it first appears. Needless to say, the 800 words in the dictionary you receive (as well as those you can add after you are satisfied that they are correct) go a long way in making coding a simple task, once the basics are understood.

DEMO PROGRAMS

The diskette furnished has over 300 sectors of programs, including the SSEDITOR. Many of these are music as well as speech. In fact a large portion of the manual is devoted to programming music (again with an editor "MUSICED") as well as speech in assembler and TSC Extended BASIC. Also included are some neat games and such. One called MATHTUTR.BAS keeps my grandkids enthralled by the hours. It is a simple program that gives the choice of addition, subtraction, division and multiplication. The user is allowed to limit the range of numbers so that it can be enjoyed by the younger as well as the more advanced. It presents the problem on the CRT screen and also speaks the problem at the same time. If the answer is correct a congratulation type reply is given by the computer, of which there are several. If the answer is not correct a firm but polite reply from the computer is given to the player and another attempt is called for. The computer being able to speak holds their attention far longer than just a plain CRT display. I honestly feel that speech could do more for CAI "Computer Assisted Instruction" than any one other single aspect of the entire learning process, as applies to CAI.

Other programs furnished are HI-LO (game) Waveform Speech (requires a separate D/A converter JPC type) this as explained previously is by far the highest fidelity method available, but is hard on memory. Also are some music demonstrations, such as

MINUET IN G	4 Voice
REGARDS TO BROADWAY	4 Voice
REGARDS TO BROADWAY	1 Voice
DARLING CLEMENTINE	1 Voice
TOP OF OLD SMOKEY	1 Voice

The music editor makes programming music easy. The music editor furnished with the board is a single part editor. A 4 part editor is available but we did not receive it for review so I can not tell you at this time how it actually works. Even the single voice editor allows "stereo" music, however two boards or one of their SING "N" STEREO boards are required. As this review is done on the SS-1 board, again I can not tell you much about the others.

A special program "PLAY" is included, this allows all binary music to be played as a command function in FLEX. Also included are several "sound effect" programs; STORM, SIREN, MORSE, PLANE, PHASOR. Some in assembler, with source or BASIC.

Additional programs furnished allow speech data developed in the SSEDITOR and saved to disk to be spoken by the computer by the TALK command. Others are TALKFIFO which uses a FIFO buffer of 64 or less phonemes, the entire string can be stored in less than 1 millisecond. This frees up processor time.

TALKIRQ allows the system to process speech in an

Interrupt system. The manual includes instructions on hardware configuration for the board's PIA.

TALKVARY allows words to be spoken and their speed controlled for demonstration purposes.

TALKFILE allows the users to listen to a phoneme file previously saved out of the editor. In other words, once a speech file is saved to disk it can be spoken by the computer by simply calling TALKFILE and the file.

SETSPEED allows close calibration of phonemes.

ALLPHNS reproduces all 64 phonemes.

HARDWARE

The SS-1 board has full capability for speech, music and sound effects. Actually it is two circuits. One a digital to analog converter (DAC) is driven by the A side of a PIA, and the other is a phoneme speech synthesizer driven by the B side of the PIA.

The output of the DAC ranges from 0 to 5 volts (reference and supply voltage). It is considered an 8 bit converter as it is attached to 8 lines of the PIA, thus allowing 256 different voltage levels in the range 0-5 volts. As mentioned previously "waveform" speech will require an additional AD converter, such as the JPC AD-16 unit (see JPC advertising - 68 Micro Journal). The FIFO buffer is composed of two ICs 334) 64 byte FIFO, the phoneme chip an interface circuit and the driver (PIA). The FIFO buffer holds 8 seconds of speech. Additional time can be had by using the computers memory and the Interrupt programs provided.

All 8 lines of the "B" side of the PIA are used to drive the speech synthesizer. Six of these lines encode one of the 64 phonemes. The remaining two encode the inflection scale (1-4).

The remainder of the circuitry does the audio summing and amplification (2 watts nominal).

Three pots are mounted on the top of the board for easy access, they control: volume, balance and the master clock frequency of the speech synthesizer.

A five pole DIP switch is mounted on the board to enable or disable several functions of the board. The functions affected are the D/A portion as to its control of the speech speed. This switch position allows simultaneous play back of speech and music. The remainder of the switch controls the interrupt line of the host computer system, one for each side of the PIA. Also the switch allows the PIA to control the NMI line of the computer for both the A and B side.

ADDITIONAL SOFTWARE

Two other disks were included in our review package. They both sell for \$29.95 each and contain games, music (4 voice, which is very pleasing) and special sound effects. Source is included for both binary and BASIC programs. They are:

SF-1 Optional Software

Consisting of 25 different programs.

SF-2 Optional Software

Consisting of 18 additional programs.

While space constraints will not allow a detailed discussion of the programs contained on the two additional diskettes it is our hearty suggestion that they be purchased with the board as they not only have a wealth of programs but the source listings are valuable in learning the more advanced processes in fully using

the system.

Additional information concerning the SS-1 SPEAK "N" SING board can be secured from:

SPEECH SYSTEMS
38 W. 255 Deerpath Road
Batavia, IL. 60510
(312) 879-6880

The price, Assembled and Tested, is \$219.95.

The quality of both the software and hardware is above average. We have run this system now for over a year and have not experienced any soft/hard failures. Again I regret that this review was delayed. However, it did allow us to include some functions that were not in our original package and it attests to the durability of the product.

Staff - 68 Micro Journal - - -

6800 to 6809

Converting 6800 Assembler Language
to 6809 Assembler Language

by E. M. Pass, Ph.D.
Computer Systems Consultants, Inc.
1454 Latta Lane, Conyers, GA 30207
Telephone Number 404-483-1717/4570

GENERAL

The conversion of 6800/1 assembler language programs to 6809 assembler language programs is an important topic. The discussion below attempts to structure this process and provide assistance to those attempting to do so. It will be divided into the following conversion phases:

basic conversion;
non-basic conversion;
optimization.

BASIC CONSIDERATIONS

Most 6800/1 mnemonic instructions have an identical 6809 symbolic representation and interpretation. This is to be expected, since the 6809 is intended to be generally upward-compatible from the 6800/1. These instructions thus require no further attention in terms of the basic language process.

A few 6800/1 instructions have an identical 6809 representation but non-identical interpretation. This may present difficulties in terms of the non-basic conversion considerations. These instructions should be recognized during the basic conversion process so that the problems they create may be better addressed during the later conversion process. They are the following:

ASR,LSR,ROR	6800/1 affects V flag
CMP,NEG,SBC,SUB	6800/1 clears H flag
SWI	6809 stacking order is different
TST	6800/1 clears C flag

The remaining 6800/1 instructions have different 6809 symbolic representations, or none at all. Depending upon the circumstances, it may be desirable to provide less than identical 6809 interpretation for certain instructions. These instructions are as follows:

ABA	PSHS B ADDA ,S+
ASLD	ASLB ROLA
CBA	PSHS B CMPA ,S+
CLC	ANDCC \$FE
CLI	ANDCC \$EF
CLV	ANDCC \$FD
CLZ	ANDCC \$FB
CPX _	CMPS _
DES	LEAS -\$01,S
DEX	LEAX -\$01,X
INS	LEAS \$01,S
INX	LEAX \$01,X
LDAA _	LDA _
LDAB _	LDB _
LSRD	LSRA RORB
PSHA	PSHS A
PSHB	PSHS B
PSHX	PSHS X
PULA	PULS A
PULB	PULS B
PULX	PULS X
SBA	PSHS B SUBA ,S+
SEC	ORCC \$01
SEI	ORCC \$10
SEV	ORCC \$02
SEZ	ORCC \$04
STAA _	STA _
STAB _	STB _
TAB	TFR A,B TSTA
TAP	TFR A,CC
TBA	TFR B,A TSTA

TPA	
TST	TFR CC,A
	TST
TSTA	ANDCC \$FE
	TSTA
TSTB	ANDCC \$FE
	TSTB
TSX	ANDCC \$FE
	TFR S,X
TXS	
	TFR X,S
WAI	
	CWAI \$FF

Note that the TST instruction is included in this table to indicate its 6809 symbolic representation. The other 6800/1 instructions with identical 6809 representations are not normally provided additional processing.

NON-BASIC CONSIDERATIONS

The conversions discussed in the previous section were generally of a rather clerical nature. For many programs, no further work is necessary to complete the conversion. However, due to non-identical 6809 interpretations of some of the 6800/1 instructions, the basic conversions are not sufficient in many cases, and more complex conversion procedures must be used.

These non-identical interpretations arise in several manners. One is in the non-identical instruction interpretations of the following instructions:

ASR,LSR,ROR	6800/1 affects V flag
CMP,NEG,SBC,SUB	6800/1 clears H flag
CPX	6800/1 sets only Z correctly
SWI	6809 stacking order is different
TSX	6800/1 sets X to S+1
TXS	6800/1 sets S to X-1

Although the 6809 interpretation of the 6800/1 instructions could be made identical with the 6800/1 interpretation, this is not usually done. The extra 6809 code in all cases is not considered worth the reduction in problems in a small number of cases, nor is it necessarily considered desirable. Even if the interpretations were made identical, not all of the conversion problems would be solved, due to architectural differences between the processors.

The non-basic conversion procedures generally involve re-debugging the converted code, looking for situations in which the condition code register does not contain the expected contents, the stack pointer is off-by-one, etc. Since most of these situations are associated with certain instructions or sequences of instructions, they should be generally easy to locate.

SPECIFIC PROBLEM AREAS IN CONVERSION

There are several areas of concern in the conversion which are not necessarily associated with certain instructions or sequences of instructions, and thus may cause problems in subtle manners which may be relatively difficult to locate. Several areas are discussed below.

The format of the condition code registers on the 6800/1 and 6809 are identical except for the 6809 E and F flags, the high-order two bits in the register. These bits are always 1 on the 6800/1, but not on the 6809. Either assuming that the bits are 1 or modifying them may cause processing errors.

The stacking order during interrupt and SWI processing differs on the two classes of machines. On the 6800/1, the stacking order is as follows (lower to higher addresses):

CC,B,A,XH,XL,PCH,PCL

whereas, on the 6809, the stacking order is as follows:

CC,A,B,DP,XH,XL,YH,YL,UM,UL,PCH,PCL

In addition to the five additional bytes pushed onto the 6809 stack, the order of the A and B registers is reversed from the 6800/1.

The S register on the 6800/1 points to the memory address one less than the last item pushed onto the stack, whereas the S register on the 6809 points to the memory address of the last byte pushed onto the stack. This may cause problems with the TSX and TXS instructions and with accessing parameters to subroutines. Interrupt handling code must take into account both the 6809 stacking order and the fact that the S register points to the top of the stack, not one location below.

External addresses, such as for the I/O and operating system, will probably be different between the 6800/1 and 6809 implementations.

The number of bytes required to express 6809 programs is normally larger than the number required on the 6800/1. This may cause difficulties in attempting to fit programs into specific sizes, such as in PROMs. It also may cause assembly errors due to out-of-range branches, which may be corrected by using the 6809 long branch instructions as required.

OPTIMIZATION OF 6809 CODE

Because of the larger instruction set and choices of addressing modes on the 6809, many 6800/1 programs may be significantly improved once converted to the 6809. Generally, this optimization process is performed after the converted program has been successfully debugged on the 6809. It can, however, be performed during the conversion process. The discussion below addresses some of the most common, easiest optimization procedures for converted 6800/1 programs.

Convert contiguous sequences of LEAX or LEAS instructions into the equivalent single instruction. On the 6800/1, the easiest manner in which to increment or decrement the X or S registers by a small amount is sometimes to code multiple DES, DEX, INS, INX instructions. After conversion, these become 6809 LEAS or LEAX instructions and may be combined. The relative effect on the condition-code register flags will generally be zero.

Combine LEAX or LEAS instructions with nearby or adjacent indexed instructions when the effect is a pre-decrement or post-increment index operation by one or two. The effect on the condition-code register flags must be considered, as the index modification will then no longer affect the condition-code register.

Combine sequences of instructions involving the A and B registers into the equivalent sequence involving the D register, when possible. Often on the 6800/1, load, store, compare, add, subtract, and other operations operate on 16-bit fields between memory and the A and B registers. On the 6809 these 8-bit operations can often be replaced by true 16-bit operations between memory and the D register. The condition-code register contents will almost certainly be different after the combination of instructions and must be considered.

Combine contiguous sequences of PSHS or PULS operations into a smaller number of PSHS or PULS operations, when possible. Regardless of the stated

order of registers in the 6809 assembly representation of an instruction, registers are always pushed and pulled (in a given 6809 instruction) to maintain the stacking order described above, although not necessarily all of the registers will be stacked or unstacked.

A sequence of PULS instructions may be combined with a following RTS instruction by appending ",PC" to the end of the last resultant PULS instruction. A pull of a register followed immediately by a push of the same register may often be replaced by a load of that register from the top of the stack (,S), although the condition-code register contents will be modified by the load, but not by the pushes and pulls. A push of a register, followed by a pull of a different (but same size) register may be replaced by a transfer from the first register to the second, without changing the contents of the condition code register.

Look for longer sequences of instructions, especially those involving movement or comparison of contiguous bytes in memory, which may be replaced by 6809 code which uses the Y and U registers to avoid extra saves and restores of the X register. Especially in time-critical areas, this class of change has the potential for more substantial improvement of 6809 code than does the other areas, although it may be more difficult.

As is the case with any program change, the optimized program must be thoroughly retested.

SUMMARY

The discussion above attempted to structure this process and provide assistance to those attempting to do so. It was structured in the following manner:

basic conversion;
non-basic conversion;
optimization.

CONTINUED FROM LAST MONTH

Home Acct Prog

Part III

ERNEST STEVE WATSON
11701 ST. CHARLES BLVD.
LITTLE ROCK, ARKANSAS 72211

OR

F. DALE BRADY
7729 BRADLEY DRIVE
LITTLE ROCK, ARKANSAS 72209

```
0 REM THIS PROGRAM READS IN FROM A DATA FILE
  INFORMATION
20 REM CON ERMING THE PROPOSED BUDGET FOR ALL INCOME
  AND
30 REM EXPENSE ACCOUNTS DURING A ONE YEAR PERIOD. IT
40 REM THEN COMPARES A PRORATED BUDGETED AMOUNT WITH
50 REM THE ACTUAL EXPENSES INCURRED YEAR-TO-DATE,
  PRINTING
60 REM THE CURRENT DIFFERENCE AND THE PROJECTED
  AMOUNT
70 REM OF SUCH DIFFERENCE FOR THE YEAR.
75 DIGITS 5,0
80 PRINT "BUDGET PROGRAM"
90 PRINT
100 PRINT "ENTER THE NAME OF THE GENERAL LEDGER FILE"
110 PRINT "FROM WHICH THE COMPUTATIONS WILL BE MADE"
120 INPUT "FOR EXAMPLE 'GL10B1.DAT'",GL$
130 OPEN OLD GL$ AS I
```

```
140 PRINT
150 PRINT "ENTER THE NAME OF THE BUDGET DATA FILE"
160 INPUT "FOR EXAMPLE 'BUDGETB1.DAT'",BF$
170 OPEN OLD BF$ AS 2
180 PRINT "ENTER THE EXPIRED AMOUNT OF THE YEAR TO
  DATE"
190 INPUT "AS A TOTAL OF DAYS",P
200 P=P/365
210 ON ERROR GOTO 600
220 INPUT #2,A,A$,B
230 X=X+1
240 GOTO 220
250 CLOSE 2
260 OPEN OLD BF$ AS 2
270 DIM A(X),A$(X),B(X)
280 FOR I=1 TO X
290 INPUT #2,A(I),A$(I),B(I)
300 NEXT I
310 CLOSE 2
320 ON ERROR GOTO 620
330 DIM B1(X)
340 INPUT #1,A1,A1$,B1
350 IF A1<400 THEN GOTO 340
360 FOR I=1 TO X
370 IF A1=A(I) THEN B(I)=B(I)+P:B(I)-B(I):GOTO 340
380 NEXT I
390 CLOSE 1
400 PRINT "A/C":TAB(4);
410 PRINT "ACCOUNT":TAB(25);
420 PRINT "BUDGET":TAB(32);
430 PRINT "ACTUAL":TAB(39);
440 PRINT "OVER/":TAB(45);
450 PRINT "PROJECTED"
460 PRINT "NO.":TAB(4);
470 PRINT "NAME":TAB(21);
480 PRINT "AMOUNT":TAB(32);
490 PRINT "AMOUNT":TAB(39);
500 PRINT "UNDER":TAB(45);
510 PRINT "FOR YEAR "
520 PRINT
522 V$="#####"
524 Z$=CHR$(92)+*+CHR$(92)
530 FOR I=1 TO X
535 IF A(I)<400 THEN GOTO 580
537 IF B1(I)<0 THEN B1(I)=B1(I)+(-I)
538 IF B(I)<0 THEN B(I)=B(I)+(-I)
540 PRINT A(I):TAB(4);
550 PRINT USING V$,A$(I):TAB(25);
560 PRINT USING V$,B(I):TAB(32);
565 B4=B4+B(I)
570 PRINT USING V$,B1(I):TAB(39);
571 B2=B2+B1(I)
572 PRINT USING V$,B(I)-B1(I):TAB(45);
573 B6=B6+(B1(I)-B(I))
574 PRINT USING V$, (1/P)*(B1(I)-B(I))
576 B8=B8+((1/P)*(B1(I)-B(I)))
580 NEXT I
582 PRINT
583 PRINT TAB(5);"TOTALS";
584 PRINT TAB(25);
585 PRINT USING V$,B4:TAB(32);
586 PRINT USING V$,B2:TAB(39);
587 PRINT USING V$,B6:TAB(45);
588 PRINT USING V$,B8
590 END
600 IF ERR<>0 THEN ON ERROR GOTO 0
610 RESUME 250
620 IF ERR<>0 THEN ON ERROR GOTO 0
630 RESUME 390

0 REM BALANCE.BAS
15 REM 1/11/82
20 CL$=CHR$(27)+*E*
30 MZ=60
40 PRINTCL$
50 PRINT:PRINT
60 REM GET YZ=YEAR,P$=GENLE6 MONTH,M$=CURRENT MONTH
70 OPEN"1.YEAR"AS1:GET#1,RECORD1
80 FIELD#1,2ASTN$,2ASCY$,3ASCMS$,3ASPM$,4OASCN$
90 YZ=CVT$(CY$):P$=PM$:M$=CM$:N$=CN$
100 CLOSE1
```

```

110 REM GET SIZE OF GENLEG FILE AND DIM VAR.
120 OPENOLD*!.*+P*+.GL* AS1
130 GET#1,RECORD1:FIELD#1,2ASZ$:XZ=CVT$(Z$)
140 DIMNZ(XZ),A$(XZ),A(XZ)
150 REM READ IN GEN LEG
160 GET#1,RECORD1:GOTO180
170 GET#1
180 FORSZ=0T07
190 FIELD#1,SZ#30ASZ$,2ASGN$,20ASGM$,BASGT$
200 IFGM$="ORCVT$(GN$)=0THEN240
210 IZ=IZ+1
220 NZ(IZ)=CVT$(GN$):A$(IZ)=GM$:A(IZ)=CVT$(GT$)
230 IFIZ=XZTHEN260
240 NEXTSZ
250 GOTO170
260 PRINTCL$
270 PRINT:PRINT
280 PRINTTAB(25);N$
290 PRINT:PRINTTAB(25);N$;" 1,";YZ
300 PRINT:PRINT
310 PRINT"ASSETS"
320 PRINT
330 FORIZ=1TOXZ
340 IFNZ(IZ)>=200THEN390
350 IFA(IZ)=0THEN380
360 GOSUBB70
370 TA=TA+A(IZ)
380 NEXTIZ
390 REM
400 A$="TOTAL ASSETS":A=TA
410 GOSUBB90
420 GOSUB900
430 PRINT:PRINT"LIABILITIES"
440 PRINT
450 FORIZ=1TOIZ
460 IFNZ(IZ)>=300THEN510
470 IFA(IZ)=0THEN500
480 GOSUBB70
490 TL=TL+A(IZ)
500 NEXTIZ
510 A$="TOTAL LIABILITIES":A=TL
520 GOSUBB90
530 PRINT:PRINT"EQUITY"
540 PRINT
550 FORIZ=1TOIZ
560 IFNZ(IZ)>=400THEN610
570 IFA(IZ)=0THEN600
580 GOSUBB70
590 TE=TE+A(IZ)
600 NEXTIZ
610 A$="TOTAL EQUITY":A=TE
620 GOSUBB90
630 A$="TOTAL LIAB & EQUITY":A=TL+TE
640 GOSUB900
650 PRINT:PRINT:FORI=1TO(WZ-3):PRINT" ";NEXTI
660 PRINT:PRINT"EXPENSE"
670 PRINT
680 FORIZ=1TOIZ
690 IFNZ(IZ)>=500THEN740
700 IFA(IZ)=0THEN730
710 GOSUBB70
720 TI=TI+A(IZ)
730 NEXTIZ
740 A$="TOTAL EXPENSES":A=TI
750 GOSUB900
760 PRINT:PRINT"INCOME"
770 PRINT
780 FORIZ=1TOIZ
790 IFNZ(IZ)>=600THEN840
800 IFA(IZ)=0THEN830
810 GOSUBB70
820 TI=TI+A(IZ)
830 NEXTIZ
840 A$="TOTAL INCOME":A=TI
850 GOSUB900
860 END
870 PRINTUSING"1234567890123456789\
      $$$$,$$,$$,$$,A$(IZ),A(IZ)
880 RETURN
890 PRINTTAB(3);A$;TAB(35);PRINTUSING"$$$,$$,$$,$$,A

```

```

:RETURN
900 PRINT
910 PRINTTAB(10);FORX=1TO LEN(A$):PRINT" ";
:NEXTX:PRINT
920 PRINTTAB(10);A$;TAB(46);
:PRINTUSING"$$$,$$,$$,$$,A
930 PRINTTAB(10);FORX=1TOLEN(A$):PRINT" ";
:NEXTX:PRINT
940 PRINT:PRINT
950 RETURN

0 REM EDITGL.BAS
20 CL$=CHR$(27)+*E*
24 PRINTCL$
25 PRINT:PRINT:PRINT
26 PRINT"THIS PROGRAM CAN BE USED TO CHANGE ACCOUNT"
27 PRINT"NAME OR NUMBER, ALSO LIST ACCOUNTS"
30 W=60
40 REM GET Y=YEAR,GL$=GENLEG MONTH,M$=CURRENT MONTH
50 OPENOLD*1.YEAR*AS1:GET#1,RECORD1
60 FIELD#1,2ASTN$,2ASTY$,3ASTM$,3ASTG$,40ASN$
70 N$=N$
80 Y=CVT$(TY$):GL$=TM$:M$=TM$
90 CLOSE1
100 REM GET SIZE OF GENLEG FILE AND DIM VAR.
110 OPENOLD*1.*+GL$+.GL* AS1
120 GET#1,RECORD1:FIELD#1,2ASZ$:XZ=CVT$(Z$)
130 DIMNZ(XZ),A$(XZ),A(XZ)
140 REM READ IN GEN LEG
150 GET#1,RECORD1:GOTO170
160 GET#1
170 FORSZ=0T07
180 FIELD#1,SZ#30ASZ$,2ASGN$,20ASGM$,BASGT$
190 IFGM$="ORCVT$(GN$)=0THEN230
200 IZ=IZ+1
210 NZ(IZ)=CVT$(GN$):A$(IZ)=GM$:A(IZ)=CVT$(GT$)
220 IFIZ=XZTHEN250
230 NEXTSZ
240 GOTO160
250 PRINTCL$
260 PRINTTAB(W/2-12);"Edit the General Ledger"
270 PRINT:PRINTTAB(W/6);"This information from ";
GL$;".GL file."
280 PRINT
290 PRINTTAB(W/4);"1. Change account NUMBER"
300 PRINTTAB(W/4);"2. Change account NAME"
310 PRINTTAB(W/4);"3. List of ACCT#/NAME"
320 PRINTTAB(W/4);"4. Return to MENU"
330 PRINT
340 PRINTTAB(W/4);"?? YOUR CHOICE (1-4)? ";
A$=VAL(INCH$(0)):PRINT
350 IFA(1 OR A):4THEN340
360 ONAGOTO450,630,800,790
370 REM ADD TO GEN LEG FILE
380 RZ=IZ/8:SZ=(Z-(RZ*8))
390 GET#1,RECORDRZ+1
400 FIELD#1,SZ#30ASZ$,2ASGN$,20ASGM$
410 LSETGM$=CVT$(NZ(IZ))
420 LSETGM$=A$(IZ)
430 PUT#1,RECORDRZ+1
440 RETURN
450 REM CHANGE ACC#
460 PRINTCL$
470 PRINTTAB(W/2-8);"CHANGE ACCOUNT NUMBER"
480 PRINT:PRINT
490 PRINT"Enter END to return to MENU"
500 INPUT"Enter Account Number to be changed",AN$
510 IFAN$="END"THEN250
520 NZ=VAL(AN$):IFNZ<100 OR NZ>900THEN
PRINTCHR$(7);"NUMBER OUT OF RANGE":GOTO480
530 FORIZ=1TOXZ:IFNZ=NZ(IZ)THEN540ELSENEXTIZ
:PRINTCHR$(7);"THAT ACCOUNT NUMBER NOT FOUND"
:GOTO480
540 PRINT:PRINT"ACCO ";NZ(IZ);TAB(15);A$(IZ)
550 PRINT:PRINT"IS THIS CORRECT ACCOUNT (Y/N)? ";
AN$=INCH$(0):PRINT
560 IFAN$<>*Y*THENPRINTCHR$(7):GOTO450
570 INPUT"Enter NEW ACC# (END TO RETURN)",AN$
580 IFAN$="END"THEN450
590 NZ=VAL(AN$):IFNZ<100 OR NZ>900THENPRINT"NUMBER

```

```

      OUT OF RANGE":GOTO570
600 FOR IZ=1 TO IZ: IF NZ=NZ(IZ) THEN 610 ELSE NEXT IZ
      :GOTO620
610 PRINT CHR$(7); "THAT ACC# IN USE":GOTO570
620 NZ(IZ)=NZ:GOSUB370:GOTO450
630 REM CHANGE ACCT NAME
640 PRINT CL$
650 PRINT TAB(W/2-8); "CHANGE ACCOUNT NAME"
660 PRINT:PRINT
670 PRINT "Enter END to return to MENU"
680 INPUT "Enter Account Name (or Number) to
      change", AN$
690 IF AN$="END" THEN 250
700 NZ=VAL(AN$)
710 FOR IZ=1 TO IZ: IF NZ=NZ(IZ) OR AN$=A$(IZ) THEN 720
      ELSE NEXT IZ: PRINT CHR$(7); AN$; " CANNOT BE FOUND"
      :GOTO660
720 PRINT:PRINT "ACCT# "; NZ(IZ); TAB(15); A$(IZ)
730 PRINT:PRINT "IS THIS CORRECT ACCOUNT (Y/N)? ";
      : NZ=INCH$(0):PRINT
740 IF AN$(">Y") THEN PRINT CHR$(7):GOTO630
750 PRINT:INPUT "Enter NEW ACCOUNT NAME", AN$
760 IF VAL(AN$)(">0") THEN PRINT CHR$(7):GOTO750
770 A$(IZ)=AN$
780 GOSUB370:GOTO630
790 CLOSE 1:CHAIN "MENU.BAS"
800 REM LIST
810 PRINT CL$
820 PRINT TAB(W/2-8); "LIST OF ACCOUNTS"
830 PRINT:PRINT "DO YOU WANT PRINTER OUTPUT (Y/N)? ";
      : AN$=INCH$(0)
840 PRINT
850 IF AN$="Y" THEN POKE40972,0:EXEC, "TTYSET,PS=N,NL=25"
860 PRINT "LIST OF ACCOUNTS FOR "; NZ$
870 PRINT:PRINT:PRINT
880 FOR IZ=1 TO IZ
885 IF IZ=1 THEN 900
890 PRINT USING "###.###.##" \234567890123456789\
      $$$,###.##", NZ(IZ), A$(IZ), A(IZ)
900 NEXT IZ
910 IF AN$="Y" THEN POKE40972,1:EXEC, "TTYSET,PS=Y,NL=0"
920 PRINT "HIT ANY KEY TO RETURN"; AN$=INCH$(0)
930 GOTO250

```

```

0 REM DELGL.BAS
20 CL$=CHR$(27)+ "E"
30 N=60
40 PRINT CL$
50 PRINT:PRINT TAB(W/2-16); "DELETE ACCOUNT IN GENERAL
      LEDGER"
60 PRINT:PRINT
70 GOSUB700
80 PRINT
90 OPENOLD "1.YEAR" AS 1:GET#1,RECORD1
100 FIELD#1,7ASZ$,3ASPM$
110 GL$=PM$
120 CLOSE 1
125 PRINT:PRINT "DELETIONS WILL BE TO THE ";GL$;"
      GENERAL LEG FILE."
130 REM GET SIZE OF GENLEG FILE AND DIM VAR.
140 OPENOLD "1." + GL$ + ".L" AS 1
150 GET#1,RECORD1:FIELD#1,2ASZ$:IZ=CVT$(Z$)
160 DIM NZ(IZ), A$(IZ), A(IZ)
170 REM READ IN GEN LEG
180 GET#1,RECORD1:GOTO200
190 GET#1
200 FORSZ=0 TO 7
210 FIELD#1,SZ:30ASZ$,2ASGN$,20ASGM$,BASGT$
220 IF GM$=" " OR CVT$(GN$)("<100") THEN 260
230 IZ=IZ+1
240 NZ(IZ)=CVT$(GN$):A$(IZ)=GM$:A(IZ)=CVT$(GT$)
250 IF IZ=IZ THEN 280
260 NEXTSZ
270 GOTO190
280 CLOSE 1
290 PRINT:PRINT

```

```

300 PRINT "Enter END to return to MENU"
310 INPUT "Enter Acc# to DELETE ", AN$
320 IF AN$="END" THEN 500
330 NZ=VAL(AN$)
340 IF NZ=100 OR NZ=900 THEN PRINT CHR$(7); "NUMBER OUT OF
      RANGE":GOTO310
350 FOR IZ=1 TO IZ: IF NZ=NZ(IZ) THEN 370 ELSE NEXT IZ
360 PRINT CHR$(7); "ACCOUNT #"; NZ; "NOT FOUND (REENTER)"
      :PRINT:GOTO300
370 PRINT:PRINT
380 PRINT USING "###.###.##" \1234567890123456789\
      $$$,###.##", NZ(IZ), A$(IZ), A(IZ)
385 IF A(IZ)(">0") THEN PRINT:PRINT CHR$(7); "ERROR ACCOUNT
      HAS BALANCE AND CANNOT BE DELETED":PRINT "HIT
      ANY KEY TO CONTINUE"; AN$=INCH$(0):AN$="N":GOTO400
390 PRINT:PRINT "IS THIS CORRECT (Y/N)? ";
      : AN$=INCH$(0)
400 IF AN$="N" THEN PRINT CL$:GOSUB700:GOTO290
410 IF AN$(">Y") THEN PRINT CHR$(7):GOTO390
420 PRINT
430 PRINT "ARE YOU SURE (Y/N)? "; AN$=INCH$(0)
440 IF AN$(">Y") THEN PRINT CL$:GOSUB700:GOTO 90
450 IF IZ=IZ THEN 480
460 NZ(IZ)=NZ(IZ+1):A$(IZ)=A$(IZ+1):A(IZ)=A(IZ+1)
470 IZ=IZ+1:GOTO450
480 IZ=IZ-1
490 PRINT CL$:GOSUB700:GOTO 90
500 PRINT "UPDATING GENERAL LEDGER FILE"
510 OPENNEW "1.DUMP.GL" AS 1
520 RZ=0
530 FORSZ=0 TO 7
540 FIELD#1,SZ:30ASZ$,2ASGN$,20ASGM$,BASGT$
550 LSETGN$=CVT$(NZ(RZ+SZ))
560 LSETGM$=A$(RZ+SZ)
570 LSETGT$=CVT$(A(RZ+SZ))
580 IF RZ+SZ=IZ THEN PUT#1:GOTO620
590 NEXTSZ
600 PUT#1
610 RZ=RZ+8:GOTO530
620 GET#1,RECORD1:REM ADD FILE SIZE
630 FIELD#1,2ASZ$:LSETZ$=CVT$(IZ)
640 PUT#1,RECORD1
650 CLOSE 1
660 KILL "1." + GL$ + ".GL"
670 RENAME "1.DUMP.GL", "1." + GL$ + ".GL"
680 PRINT:PRINT "LOADING MENU"
690 CHAIN "MENU.BAS"
700 PRINT TAB(12); "*****WARNING*****"
710 PRINT "IF ACCOUNTS ARE DELETED, TRANSACTIONS FOR"
720 PRINT "DELETED ACCOUNTS CANNOT BE POSTED TO
      GENERAL"
730 PRINT "LEDGER. PLEASE DON'T DELETE ACCOUNTS
      UNTIL"
740 PRINT "THOSE TRANSACTIONS RE POSTED."
750 RETURN

```

```

230 IF IZ=IZ THEN 260
240 NEXTSZ
250 GOTO170
260 PRINT
280 PRINT FRE(0)

```

```

0 REM B LANCE.BAS
30 CL$=CHR$(27)+ "E"
40 NZ=60
43 PRINT CL$
44 PRINT:PRINT
45 PRINT TAB(WZ/2-7); "Balance Sheet"
50 REM GET YZ=YEAR, P$=GENLEG MONTH, M$=CURRENT MONTH
60 OPEN "1.YEAR" AS 1:GET#1,RECORD1
70 FIELD#1,2ASTN$,2ASCY$,3ASCM$,3ASPM$,40ASCN$
90 YZ=CVT$(TY$):P$=PM$:M$=CM$:N$=CN$
100 CLOSE 1
110 REM GET SIZE OF GENLEG FILE AND DIM VAR.
120 OPENOLD "1." + P$ + ".GL" AS 1
130 GET#1,RECORD1:FIELD#1,2ASZ$:IZ=CVT$(Z$)
140 DIM NZ(IZ), A$(IZ), A(IZ)
150 REM READ IN GEN LEG
160 GET#1,RECORD1:GOTO180
170 GET#1
180 FORSZ=0 TO 7
190 FIELD#1,SZ:30ASZ$,2ASGN$,20ASGM$,BASGT$
200 IF GM$=" " OR CVT$(GN$)(">0") THEN 240
210 IZ=IZ+1
220 NZ(IZ)=CVT$(GN$):A$(IZ)=GM$:A(IZ)=CVT$(GT$)
225 PRINT NZ(IZ), A$(IZ), A(IZ)

```

TO BE CONTINUED

ET3400

BASIC FOR A MODIFIED ET-3400/ETA-3400 MICROCOMPUTER

Dr. Allen H. Wolach
Department of Psychology
Illinois Institute of Technology
Chicago, Illinois 60616

INTRODUCTION

A number of articles have included directions for interfacing the ET-3400/ETA-3400 microcomputer to SS-50 memory boards. An expanded microcomputer system needs appropriate software. The present article shows how to add a floating point arithmetic BASIC (as opposed to TINY BASIC) interpreter to the ET-3400/ETA-3400 microcomputer system. The version of BASIC that I modified for for the ET-3400/ETA-3400 microcomputer is available in a book entitled: BEST of Interface Age, Volume 1, Software in BASIC, Carl D. Warren (Ed.), Dilithium Press, P. O. Box 32, Forest Grove, Oregon 97166. The book sells for under \$15.00. The BASIC interpreter was written by Robert Uiterwyk and is on pages 240 through 278. I should note that the description of the program in the book has a number of incorrect references to memory locations (especially pages 235 and 236). This should not prove to be a problem because the modifications that are in this article are keyed to the actual program on pages 240 through 278. The program listing does not contain errors.

MEMORY MAP FOR MODIFIED ET-3400/ETA3400 MICROCOMPUTER

Modifications of the ET-3400/ETA3400 system do not always produce the same memory map. One has the option of leaving the ET-3400/ETA-3400 Heath/Wintek monitor just above the 4k of RAM in the original system (an inconvenient location for an expanded system), or moving the monitor program to a higher location. If one moves the monitor program, he must burn a new PROM for the modified monitor. I opted to leave the monitor program at its original location and modify the cassette software (such as Uiterwyk's BASIC) to accommodate the inconveniently located monitor program. I also opted to remove the 2k of ROM containing TINY BASIC (which is not necessary when one has a floating point BASIC) and replace this ROM with 2k of RAM. This ROM to RAM conversion has been described in Popular Electronics (Wolach, A. H., Simple memory addition for training computers, Popular Electronics, 1981, 19, 61-62). Uiterwyk's BASIC requires somewhat over 4k of memory. Most of the machine language interpreter program can be placed in the first 4k of the ET-3400/ETA-3400 memory map. The remainder of the program can be placed in the RAM locations previously reserved for TINY BASIC. One could place the last portion of the program in any other consecutive memory locations. A user's BASIC program must be in consecutive memory locations. I started these locations at 4000 (all memory locations are in hex) and installed a 32k SS-50 memory board with consecutive memory locations starting at 4000. The details of my 32k memory addition will appear in one of the 1982 issues of Popular Electronics. The 68' Micro Journal has

included a somewhat different SS-50 memory addition (Klem, G. H., ET/ETA-3400 to SS50, 68' Micro Journal, September, 1981, 18-20).

Suppose that one has an ET-3400/ETA3400 system with RAM in the first 4k locations. Almost all ET-3400/ETA-3400 systems will have RAM in these locations because the unmodified microcomputer system has RAM in the first 4k of memory. Start by entering Uiterwyk's BASIC interpreter (starting on page 240 of the book) until all memory locations through 0FF6 are entered. One does not have to enter the entire program at one time. A portion of the machine language can be entered and then saved on cassette until you verify that your most recent (most complete) cassette works properly.

MAKING MONITORS COMPATIBLE

Table 1 shows the changes that have to be made to make the Uiterwyk program compatible with the

Table 1. Making Monitors Compatible

Memory Location	Change Contents to	
0271	16	160A is the Heath/Wintek monitor equivalent of MIKBUG OUT2H (located at E0BF in MIKBUG)
0272	0C	

0274	16	160A is the Heath/Wintek monitor equivalent of MIKBUG OUT4HS (located at E0C8 in MIKBUG)
0275	0A	
0279	18	1865 is the Heath/Wintek monitor equivalent of MIKBUG OUTEEE (located at E1D1 in MIKBUG)
027A	65	
027C	18	18E1 is the Heath/Wintek monitor equivalent of MIKBUG INEEE (located at E1AC in M KBUG)
027D	E1	
07DD	40	Location for beginning of user BASIC program storage
07DE	00	
08E3	14	Location for return to monitor (08E3 and 08E4 would contain E0E3 for a return to MIKBUG)
08E4	00	

Heath/Wintek monitor. The changes assume that the Heath/Wintek monitor remains in its original location. If the monitor is moved to higher memory locations, the new locations must be accessed by the program. For example, one would find new memory locations that correspond to former memory locations 160C, 160A, 1865, 18E1, and 1400. Memory locations 021 and 022 would contain the new monitor equivalent of 16, and 0C, respectively. Note that locations 0DD and 07DE contain 40, and 00, respectively. I selected 4000 as the starting address of a user BASIC program. Locations 07DD and 07DE can contain any address as the starting address for BASIC, provided this address is above the memory addresses for the BASIC interpreter.

Note that Table 1 gives the Heath/Wintek monitor equivalent of Motorola's MIKBUG monitor routines named OUT2H, OUT4HS, OUTEEE, and INNEEE. This information is useful for adapting other software to the ET-3400/ETA3400 system. Most 6800 software assumes a M KBUG or MIKBUG compatible monitor.

STACK AND INDEX STACK

Table 2 shows the locations for the BASIC stack and

Table 2. Stack and Index Stack Locations

Memory Location	Potential Contents	
07F7	A0	Stack
07F8	45	
07FA	A0	Index Register
07FB	7F	Stack
0829	A0	Stack
082A	45	
08DA	A0	Patch Routine for Returning to BASIC
08DB	46	
08DD	A0	Stack
08DE	40	
08E0	A0	Stack
08E1	08	
0CAE	A0	Stack
0CAF	45	
0EC7	40	Beginning of User Program
0EC8	00	

index stack. Use these locations if your system has approximately 8k of memory starting at A000. This will make it possible to easily adapt other MIKBUG based software for the ET-3400/ETA-3400 system. MIKBUG assumes scratch pad RAM starting at A000. If your system memory does not extend beyond A000, change the memory locations in Table 2 to locations that are within the range of your memory, but above the range of the memory that will be used for user BASIC programs. For example, memory locations containing A045 in Table 2 could be changed to contain 8045. Similarly, memory locations containing A040, and A008 could be changed to contain 8040, and 8008, respectively.

RELOCATING LAST PORTION OF PROGRAM

Suppose that your system retains the Heath/Wintek monitor program in its original location. The end of the BASIC interpreter can be moved to any memory locations that are below the locations for the start of a user BASIC program. Assume that the locations selected for the end of the BASIC interpreter are in the memory locations formerly reserved for TINY BASIC. One must modify the interpreter by entering the program segments in Table 3. Relocate the end of the BASIC program

Table 3. Changes for Relocated Program

OFC2	7E	1CC4		JMP	BEGED
OFC5	C6	18	NEXT 5	LDA B	#18
OFC7	20	02		BRA	NEXT6 + 2
OFC9	C6	17	NEXT 6	LDA B	#17
OFCB	7E	1CDA		JMP	RELOC
1CC4	BD	0302		JSR	INDX
1CC7	DE	61		LDX	FORNOW
1CC9	EE	00		LDX	0 X
1CCB	BD	02D2		JSR	STOREX
1CCE	DE	61		LDX	FORNOW
1CD0	EE	0E		LDX	14 X
1CD2	DF	36		STX	BASLIN
1CD4	7E	0FBC		JMP	NEXT3
1CD7	7E	0FC5		JMP	NEXT5

in consecutive memory locations starting at 1CDA. Location 1CDA should contain what was in location OFDA of Uiterwyk's program. The remainder of Uiterwyk's interpreter program is placed in consecutive ascending memory locations. Of course, the end of Uiterwyk's interpreter can be relocated at any consecutive memory locations below the start of the memory for a user BASIC program. Memory locations referenced in Tables 3 and 4 (Table 4 is discussed below) would have to be changed if the end of the program was relocated at other memory locations.

Table 4 shows the final changes that are made if the end of the program is relocated as described above.

Table 4. Changes necessitated by Relocation

Memory Location	Change Contents to	Memory Location	Change Contents to
016B	1C	0A06	1E
016C	DD	0A07	02
01C2	1D	0A34	1C
01C3	B7	0A35	F2
01D0	1D	0EA0	1E
01D1	79	0EA1	80
01DF	1D	0F6E	1E
01E0	AE	0F67	80
01E6	1D	0F71	57
01E7	93	0F78	4C
01EE	1D	0F75	1E
01EF	6B	0F76	80
02B3	10	0FB4	1D
02B4	04	0FB5	37
07ED	1D	1D3C	1D
07EE	78	1D3D	51
0867	1E	1DCD	1D
0868	80	1DCE	EE

These changes enable earlier portions of the program to reference the relocated portion of the program. These changes also enable the relocated parts of the program to reference other relocated parts of the program.

USING BASIC

If the above modifications are made in the Uiterwyk BASIC program, the modified program will run with all of the Uiterwyk BASIC features except SAVE, LOAD, and APPEND. After the BASIC program is loaded from cassette storage, the user enters:

G 100 (followed by a carriage return)

That is, memory location 0100 is the cold start location for BASIC. One can leave the BASIC program and enter the monitor program by entering:

PATCH (followed by a carriage return)

BASIC can be reentered from the monitor by entering:

G 100 (followed by a carriage return)

or G 103 (followed by a carriage return)

The G 100 command (cold start) will cause any previous user BASIC program (entered before the user entered the monitor program) to be erased. The G 103 command will cause a return to BASIC without erasing a previous user BASIC program. Thus, memory location 103 is the location for a warm start.

SAVING AND LOADING PROGRAMS

One can save and load programs with the cassette system in the ET-3400/ETA-3400 system. When he has completed a program in BASIC, he can exit BASIC and enter the Heath/Wintek monitor by entering:

PATCH (followed by a carriage return)

Then the user examines memory locations 00AE and 00AF. These locations contain the location of the end of the user BASIC program. Suppose that 00AE contains 23, and 00AF contains 21. The end of the user program is at memory location 2321. The user can save his program on cassette by entering:

CTRL/T T8:location 1, location 2
(followed by a carriage return)

Location 1 is the starting address of the user BASIC program (4000 in my system). Location 2 is the memory location for the end of the user BASIC program (2321 in the above example). The user would enter:

CTRL/T T8:4000,2321
(followed by a carriage return)

For the above example, the cassette system would record a user program starting at 1F00, and ending at 2321. The user must also save the temporary storage information in memory locations 0000 through 00FF. This information is saved at another location on the cassette by entering:

CTRL/T T8:0000,00FF
(followed by a carriage return)

Suppose that one wants to enter the BASIC interpreter in the microcomputer and he wants to enter the program that he previously saved on tape. First, he would enter BASIC in the microcomputer from a cassette that contained the BASIC interpreter. Then the user would enter:

G 100 (followed by a carriage return)

The above procedure would enter and initialize BASIC. The user would then enter the monitor program by entering:

PATCH (followed by a carriage return)

Then he would place the cassette containing his BASIC program and the temporary locations in the recorder and position the tape so that the user BASIC program could be entered. The user would enter:

L8 (followed by a carriage return)

The user would then position the tape so that the temporary locations could be entered. Then he would enter:

L8 (followed by a carriage return)

Finally, the user would enter:

G 103 (followed by a carriage return)

If the user enters:

RUN (followed by a carriage return)

he can run the program that was loaded. Of course, the user can correct or modify the program before it is RUN.

UITERWYK'S BASIC

The book containing Uiterwyk's BASIC has a good description of its features (pages 278 through 300). The small interpreter has more features than one might expect. These features include floating point and exponential calculations to nine digits, arrays with one and two subscripts, etc. One can learn about the BASIC interpreter as he enters it into his ET-3400/ETA-3400 system. Most of the symbols used in the program listing are very helpful in explaining the function of a given section of the program. The text would have been improved if a better printer had been used for the program listing. Many E's look like F's in the program listing. This necessitates checking operational codes for some instructions. Mnemonic code is listed next to the operational code.

SUPPORT YOUR ADVERTISERS

BIT BUCKET

ESP Electronic Specialists, Inc.

171 South Main Street, Melick, Mass. 01760
(617) 655-1532

MICRO COMPUTER INTERFERENCE CONTROL HOT LINE

ELECTRONIC SPECIALISTS announce installation of a new
Toll Free Interference Control Hot Line. Call 1-800-225-6876
between 9 AM and 4 PM Eastern Time Monday thru Friday to
discuss micro computer interference control problems.
Experienced staff will analyze the problem situation and
make specific recommendations for control of troublesome
processor or peripheral interference.
Or, analyze system electrical interference problems with
the help of many typical situations and solutions outlined
in their FREE 40 page Interference Control Product catalog.

ELECTRONIC SPECIALISTS, INC., 171 SOUTH MAIN STREET
P. O. BOX 389, MELICK, MA 01760 Phone: 617-655-1532

MAC+EDIT 6809 FULL-SCREEN EDITOR

MAC+Edit is a full-screen cursor oriented editor
program written for the Electric Window++ video card.
Instructions are provided to configure MAC+Edit for other
video cards as well. Being a full-screen editor means
that all changes to a file being edited occur on the
screen at the cursor location, and that it is possible
to move the cursor to any location on the screen. Because
MAC+Edit uses a memory mapped video board instead of a
terminal, the changes made occur instantly. There is no
lag time as the terminal "refreshes" the screen. The
advantage of a full-screen editor over a conventional
context or line editor is that "what you see is what you
get". There is no need to print the line after a change
is made to see if it occurred. You see the change as it
happens. Another advantage to MAC+Edit over conventional
editors is that MAC+Edit supports the use of a printer
simultaneously with editing without the need for an
interrupt timer card. This means that files can be printed
while other files are edited.

MAC+Edit is not a word processing system and the price
reflects that fact. It is, however, a good general purpose
editor with features that make it usable for document
preparation as well as program development. Through the
use of "action" files, tabs and left and right margins
can be saved and retrieved. The default tabs are just
another "action" file which can be modified to suit the
user's needs. An additional utility, SHUFFLE, can be used
to copy, move, or insert parts of files into or out of
other files. The print function does page numbering,
titling, single, double, and triple spacing, and will
print up to 99 copies of the document. These features
along with the thirty-three (33) function keys make
MAC+Edit an outstanding value.

MAC+Edit requires 28K of memory, FLEX+ 9, and a
memory mapped video display board. The software is
supplied on a 5 1/4" FLEX+ diskette, and includes
the MAC+Edit user's manual. Price...\$50.00

MAC Software 484/476-5113
P.O. 1129
Ouluth GA 30136

*FLEX is a trademark of Technical Systems Consultants, Inc.
**Electric Window is a trademark of Percom Data Company, Inc.

NEWSRELEASE

DOUBLE DENSITY NOW AVAILABLE FOR THE EXORCISOR BUS

An advanced design floppy disk controller for the Exorcisor bus
has been announced by Smoke Signal Broadcasting of Westlake Village,
California. The controller, designated the DCB-4E, is an adaptation
of the company's DCB-4A board which is the de facto standard con-
troller for the 99-50 bus.

According to Project Manager, Ken Erickson, the decision to produce
a version of the board for the Exorcisor bus was made essentially
by the users of Exorcisor bus products. Many of them were pur-
chasing the DCB-4A and attaching it to the Exorcisor bus through
a wire-wrap card. Thus, there was a built-in market to existing
customers once the project was completed. Also, the customers were
assured of reliability since the basic design on the 99-50 bus had
been proven in thousands of installations.

A unique feature of the board is its ability to handle up to four
8-inch and four 5-1/4-inch floppy drives on the same controller.
Single and double density operation is software selectable. Also,
the user can specify the track stepping rate under software control.
Single or double-sided drives and single or double-track density
are also accommodated by the controller. A 1k on-board memory
buffer allows data transfer from the disk to take place independent
of main processor control; thus, allowing the main processor to
handle interrupts and other program tasks while the DCB-4E transfers
data to and from the disk.

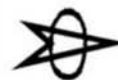
A multi-user multi-tasking operating system and supporting software
is available for the DCB-4E from Microware Systems Corporation of
Des Moines.

The price of the DCB-4E is \$695 in unit quantities. Contact Jim
Alday for more information. (213) 889-9340.

SMOKE SIGNAL

11326 Via Colinas, Westlake Village, CA 91361 • Tel (213) 889-9340

Specialty Electronics, Inc.



ATTENTION LEVEL II USERS

Specialty Electronics announces a new product for those who run
the Specialty Electronics Accounting packages. Now you don't have
to wait on the printer to be free to run your reports. General
Ledger, Accounts Receivable, Payroll, and Inventory are currently
available with a spooling option. Prices are as follows:

General Ledger.....	\$499
Accounts Receivable.....	\$399
Payroll.....	\$525
Inventory.....	\$399

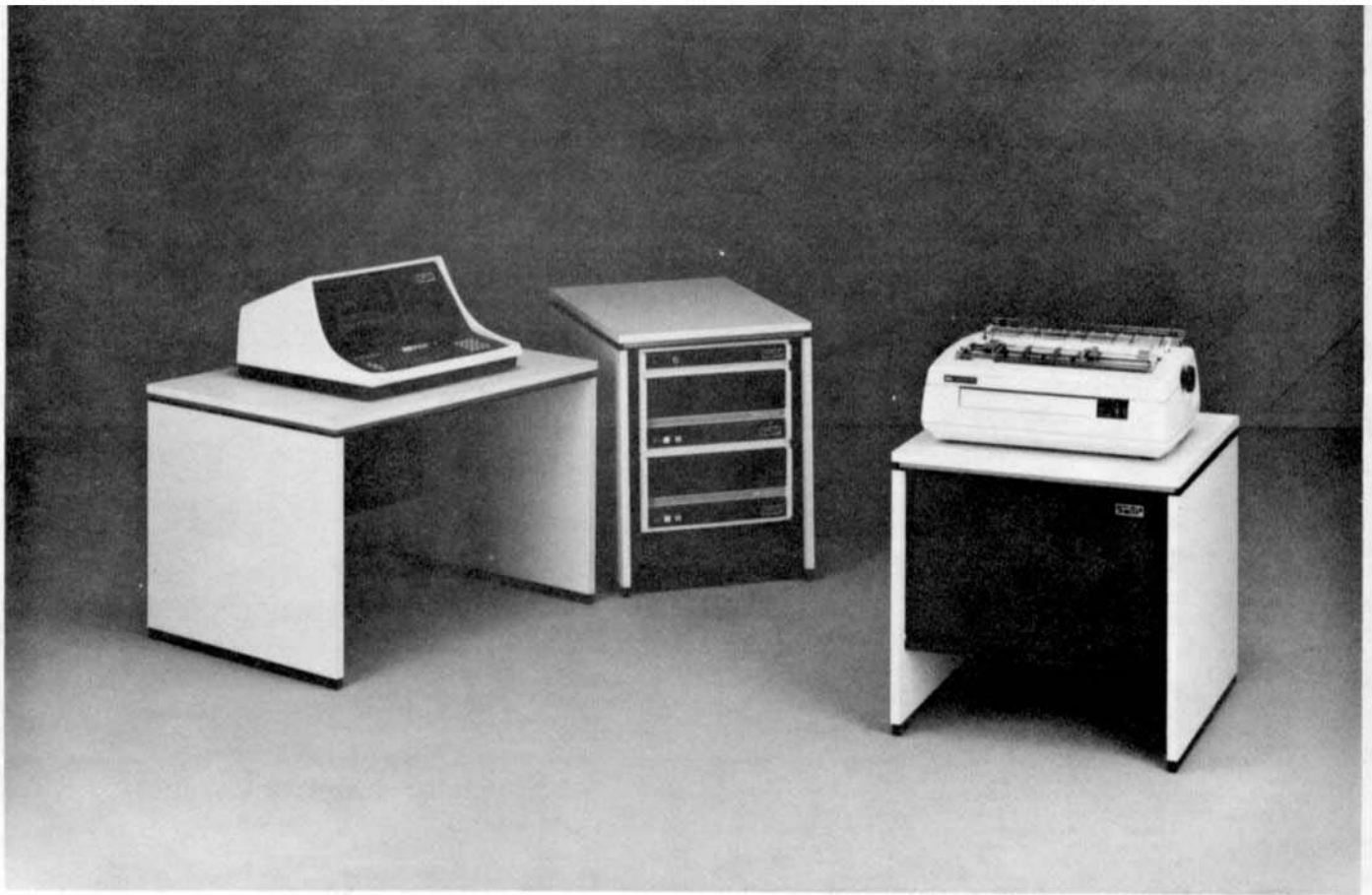
Note: For those who presently have the above packages, an upgrade
is available for \$100.

COMING SOON.....

Customer Purchase Mailing List System. This Package keeps track
of customers and their purchases for mass mailing. This package
will be available August 1, 1982 and can be purchased for \$299.
(Requires 500K disk storage, Level I or II OS9+ operating system,
132 column printer, and direct cursor address terminal)

* OS9 is a trademark of Microware Inc and Motorola Corp

(405) 233-1632 • P.O. BOX 541 • ENID, OKLAHOMA 73702



THE COMPLETE BUSINESS SYSTEM

+ Multiuser + Highly Expandable + Cost Effective

S+ THE CONCEPT

The S+ system is a modular computer system in which all portions of the hardware and software are designed to work together in the most efficient way possible. An S+ single user system with floppy disk storage is a competitive and cost effective entry level system. Unlike most other small computers being sold as "personal", or "small business" machines, the S+ system may be expanded to maximum capabilities using this same hardware and software. You cannot end up with a DEAD END system that cannot be expanded and whose software is not compatible with larger machines. A basic S+ system may be expanded to thirty-two users, a megabyte of main memory and hundreds of megabytes of hard disk storage by simply plugging in, or connecting the desired upgrade equipment.

TOTAL DESIGN—Hardware and Software

The S+ system is an integrated hardware and software design. The two complement and enhance each other in this system. The UniFLEX® operating

system used in the S+ systems is patterned after the Bell Laboratories UNIX® operating system, one of the most admired and widely used operating systems in the world. Instead of being an afterthought, the software is part of the design of the S+ system. You can be sure that with this approach that all parts of the computer operate with maximum efficiency and cost effectiveness.

THE CENTRAL PROCESSOR

The basic S+ system is configured with 256K bytes of memory and can be expanded to more than 1 million bytes. An efficient and fast hardware memory management system is used to allocate the available memory among the users on a dynamic basis. As little as 8K bytes, or the entire memory—if needed—can be used by any individual user. This makes it possible to run very large programs on the system, but it also uses no more memory than necessary for a particular job. The increase in cost effectiveness of this system over crude and outdated bank switching arrangements is dramatic.

The central processor runs in both user and supervisor states. It can detect and reject a defective user program. It is impossible for a user program to go bad and stop the entire system, as can happen quite easily in less sophisticated systems.

Task switching is accomplished by use of a multiple map RAM memory, with sixty-four individual task maps. Each task can access from 4 to 64 K-bytes of memory. Multiple tasks may be used in programs that require more than 64K bytes of memory for execution. When a task is completed the memory is automatically released for other use.

SOFTWARE

The S+ operating system, UniFLEX® is a multiuser, multitasking operating system based on the UNIX® operating system that has been used for many years on Digital Equipment Corp. PDP-11 series minicomputers. It is considered one of the most sophisticated and "user friendly" operating systems available. Variations of UNIX® are rapidly becoming standard on mini and larger microcomputers.

A large variety of languages are available for use with the system. These include FORTRAN, COBOL, BASIC, and Pascal. Word processing packages are also available to give you full text processing capability on the system.

Applications programs are available in large quantities in many fields. This includes general business, medical, dental, veterinary, library and real estate management; plus others. Since the system is multiuser it can also be connected to cash registers to produce a point-of-sale terminal system combined with the computer. The possibilities for application of this system are endless.

THE I/O SYSTEM

The S+ system is totally interrupt driven. All terminal and printer I/O devices connect to an I/O bus separate from the main bus. Up to thirty-two separate devices may be connected to the I/O bus at any one time. If I/O activity is great enough to cause an unacceptable slowdown in system operation, a separate I/O processor can be installed in the system. This plug-in option removes all I/O handling

overhead from the main processor and allows operation of up to thirty-two external devices at 9,600 baud. Without an integrated total design, as in the S+ system, it would become impractical to use a UNIX® type operating system in a situation with heavy terminal I/O activity.

DISK STORAGE

A wide range of disk storage capacity is available for the S+ system, from 2.5 M-byte floppy disks to an 80 M-byte Winchester and many sizes between. All disk controllers use direct memory access (DMA) type operations to maximize data transfer and to minimize overhead on the main processor. The Winchester disks also use intelligent controllers along with DMA transfers to preserve the performance that these type devices are capable of giving. Without this distributed intelligence the system performance would be greatly degraded. The UniFLEX® operating system is designed to work at maximum efficiency with this type disk system. The data transfer rates achieved by this combination rival those of large minicomputers.

COMMUNICATIONS

A high speed local network communications system is available to interconnect S+ systems. The VIA-BUS® network will allow communication between systems at data rates of over 400K baud. Such a system makes it possible to share data between local systems in an efficient and low-cost manner.

AVAILABLE SOON

Tape backup—20M-Byte in less than 15 minutes on a standard ¼ inch cartridge.

Mini-Wini—5 and 10 M-Byte Winchesters—5¼ inch package. Winchester performance, for smaller systems in a small package. UniFLEX® compatible design.

Large Capacity—190 and 340 M-Byte Winchesters, plus SMD cartridge drives.

UniFLEX is a registered trademark of Technical Systems Consultants, Inc.

UNIX is a registered trademark of Bell Labs.

VIABUS is a registered trademark of Southwest Technical Products Corporation.



SOUTHWEST TECHNICAL PRODUCTS CORPORATION
219 W. RHAPSODY
SAN ANTONIO, TEXAS 78216 (512) 344-0241

STAR-DOS (tm)

NEW PRODUCT ANNOUNCEMENT

We are happy to announce a major new enhancement for Radio Shack Color Computer disk systems - the STAR-DOS Disk Operating System.

STAR-DOS extends the Color Computer disk system past the limits imposed by Radio Shack's Disk Extended Basic by adding a full Disk Operating System which can be used not just by Basic programs, but by machine and assembly language programs as well. Now you can run all those fast and powerful machine language programs which cannot be used with just Basic.

The STAR-DOS Disk Operating System is a unique new concept which greatly extends the speed and power of the Color Computer disk system by blending the best of two worlds:

1. From the user's viewpoint, STAR-DOS disk format is identical with that used by standard Color Computer Disk Extended Basic. Files written by Basic can be read by machine language programs running under STAR-DOS, and vice versa. Since STAR-DOS is compatible with Basic, there is no need to buy a new Basic to go with a new DOS.

2. From the system or applications programmer's viewpoint, STAR-DOS looks just like a standard 6809 DOS. There are provisions for multiple 320-byte File Control Blocks, routines to open and close named files, read or write one byte or character at a time, rename or delete files, read or write single sectors, search or modify the directory, and more. STAR-DOS is so powerful that any machine language programs written for the popular big-system 6809 DOS can be run on the Color Computer with STAR-DOS just by changing a few addresses.

STAR-DOS is supplied on disk with a comprehensive user and programmer's manual, which explains all available routines and entry points, along with examples showing how to use them. It also comes with a number of utilities to make use of your disk system even easier and faster. It costs \$49.90 and is available NOW. Other 6809 software running under STAR-DOS (such as the All-in-One Editor/Processor and Mailing List Program) will be released soon.

Introl corp.

661 WEST VIRGINIA STREET • MILWAUKEE, WISCONSIN 53204

AREA 414/276-2837

July 1, 1982

Mr. Don Williams
68 Micro Journal
P. O. Box 849
Hixson, TN 37343

Dear Don:

The Eratosthenes' sieve benchmark test results that Norm Connors included in his July, 1982 "C User Notes" certainly helps point out the speed and efficiency of the 6809 code produced by the Introl-C compiler system. Naturally, we would like to think that Introl-C will emerge as the pace-setter in this regard when it comes to efficient, high-level languages for the 6809. However, only time will tell since several other C compilers and languages for the 6809 have yet to be heard from.

If any of your readers have run the Eratosthenes' benchmark on the 6809 using compilers other than those covered in Norm's study, we would be very interested in hearing what those results are. We have already heard from one user of TSC Pascal and Fortran 77 who ran the Pascal and Fortran Eratosthenes' programs from the September, 1981 BYTE benchmark article using a 2 Mhz 6809. On this basis, Introl-C's execution time of 11.0 seconds compares quite favorably in relation to both TSC Pascal (27 sec) and Fortran 77 (38 sec).

Also, since we will be releasing Introl-C for the UNIFLEX and OS-9 in late July, we will be especially interested in learning just how we stack up against the other C compilers currently being offered for these operating systems.

Any information your readers would be willing to share with us would be greatly appreciated.

Sincerely,

John Wisniewski
John Wisniewski

CORRECTION

Regarding an article in the August Issue, by Joseph Wicklund, concerning faults with his 6809 CPU IC, the following information applies.

Shortly after the August Issue was mailed I received a call from Motorola (guess they do read 68 Micro Journal after all) inquiring as to the problem Joseph was having with his 6809. They then contacted Joe and he in turn called me back (as did Motorola) and Joe gave me the following information.

Seems that there is a possibility that some 6809 chips, code dated - 7F7924 - had a defective mask. If you have a 6809 with this code, especially the 7F part, I would recommend that you check with Motorola, especially if you have been experiencing the type problem outlined in the August article.

I have not been able to confirm this with Motorola but no harm in checking. It is just possible that Joe's particular 6809 was defective. He reported to me that having replaced it with a newer version (code date) his problem went away.

Hope we did not send too many of you off on a 'bug' search that was not there. If I receive additional info from either Joe or Motorola will let you know.

On the whole, my personal opinion is that the 6809 is the finest 8 bit CPU ever made, even finer than a lot of 16 bitters I know of.

DMW - - -

**TELECOM INDUSTRIES CORPORATION**

27 Bonaventura Drive • San Jose, CA 95134 • (408) 282-3100

July 20, 1982

68 Micro Journal
3800 Cassandra Smith
PO Box 849
Hixson, TN 37343

Dear Don Williams Sr.,

I have been an avid reader and subscriber of the 68 Micro Journal for years now and it is one of the most valuable tools available for the 6809 computer users.

To facilitate the exchange of information among 6809 users, an exchange forum for users group (such as Heathkit's MUO) is definitely needed. The 68 Micro Journal could be an ideal center for this type of exchange. The written software for the 6809 user group could be provided by group members and offered for use by other group members at a very small charge. We all know that the 6809 offers an excellent "bone for the buck" and a user's group will help reinforce it.

As a side note - I purchased a MP-32 memory board at a very reasonable price from South East Media. The MP-32 was shipped promptly as agreed upon on the phone. I also purchased a part from Thomas Instrumentation and they shipped immediately on a phone order. The service and telephone assistance that these two companies offered, was very impressive.

Yours very truly,

Vincent M. Finelli

Vincent M. Finelli
Executive Vice President
Satellite Earth Station Division

VMF/rjt
TCOR0037

Dear Mr. Williams,

My "DETNUM" letter appeared in the June issue of 68 Micro Journal, however one page of the program listing apparently got lost. Here is the entire listing again if you would like to print it. Thanks.

Sincerely,

Randy Kron
 Randy Kron
 Rt. 2
 Keokuk, Iowa 52247

6809 RESIDENT ASSEMBLED v1.0 07/04/82 03:00:32 PM
 PAGE 002 DETNUM

```

0057 0022 EB E0      ADDB  #5      ADD IN NEW DIGIT
0058 0024 B9 00      ADCA  #0
0059 0026 E7 43      B1B      PROD-1.U
0060 0028 34 02      FDBS      A      CARRY FROM LAST OPERATION
0061 002A A6 42      LDB  PROD.U
0062 002C EA 42      LDB  PLIER.U
0063 002E 30      MUL      MSB OF PROD TIMES MULTIPLIER
0064 002F E8 E0      ADDB  #8      ADD IN CARRY FROM LSB
0065 0031 B9 00      ADCA  #0
0066 0033 E7 42      B1B      PROD.U
0067 0035 AC 41      INC  COUNT.U
0068 0037 4D      INC  STA
0069 0038 27 D8      BGTB  GETD.P
0070 003A BD E0 2A      BASHLP JBR  NATCH
0071 003D 24 FB      BCC  BASHLP
0072 003F 20 02      BNA  GETD.V
0073 0041 1C FE      DE1049 ANDDC  #0FE
0074 0043 32 61      LEAB  1.S
0075 0045 35 D4      PLAS  0.1.U,PC
0076 0047      #
0077 0047      # CONVERT HEXC1 COMPUTER IN A TO BINARY
0078 0047      # NATCH HAS ELIMINATED THE PUNCTION
0079 0047      # BETWEEN $39 AND $41 SO IF A>$39
0080 0047      # 7 IS SUBTRACTED TO CONVERT POSSIBLE
0081 0047      # HEX DIGITS
0082 0047      #
0083 0047 B0 30      CHVT  SUBA  #30      CONVERT TO BINARY
0084 0049 B1 09      CHVT  CHFA  #9
0085 004B 23 02      BLS  CHVT2
0086 004D B0 07      CHVT2 SUBA  #7
0087 004F A1 C4      CHVT2 CHFA  #PLIER.U
0088 0051 25 03      BLS  MBOX
0089 0053 1A 01      ORCC  #1
0090 0055 39      RTS
0091 0056 1C FE      ANDDC  #0FE      GOOD NUMBER
0092 0058 39      RTB
0093      END
  
```

TOTAL ERRORS 0

Extending the Life and Range of Old SS-50 Memory Boards

by E. M. (Bud) Pass, Ph.D., President
 Computer Systems Consultants, Inc.
 1454 Latta Lane, Conyers GA 30207
 Telephone Number 404-483-1717/4570

Old SS-50 memory boards may usually be quite easily modified for use with 6809 extended addressing. The change may involve as little as adding one 74S138 chip and one 1000 ohm resistor, cutting one land, and adding several jumpers. The concept of the change involves interrupting the VMA line as it enters the memory board, allowing VMA to be asserted to the remainder of the logic on the board only if the desired extended address range is present on the bus. This technique delays VMA by only a few nanoseconds and should work unless the board timing is extremely critical.

In order to use extended addressing, the system must be set up properly. If the SVTFC MP-09 CPU board is being used, the Baud-rate generator chip (MC14411) must be removed and a second 74S189 must be installed. The CPU board should also be brought up to the latest of the several modification levels. Specifically, four 470 ohm pullup resistors must be installed (if not already present) on the four output lines of the new 74S189 to regulated +5 volts, for additional noise suppression. Baud-rates must be present on the SS-30 bus (or at least on the serial boards which use them). If necessary, the mother board must be modified by separating the five Baud-rate lines on the SS-30 from those on the SS-50 bus, as the Baud-rate and extended address signals will mutually interfere.

Most operating systems for the 6809 on the SS-50 bus allow some amount of flexibility in the allocation of extended addresses to the memory boards. Essentially all systems require one or more boards configured for extended address zero for initial system boot operations. Most do not require that the other boards be physically decoded as sequentially higher addresses and will use the address translation hardware on the CPU board to logically relocate the higher-numbered boards to lower extended addresses, just as they use the translation hardware to relocate 4K blocks of addresses within the 64K non-extended address space.

Many CPU and mother boards have no extended address decoding and thus respond to sixteen-bit addresses with no regard to the extended address, mirroring CPU board and I/O functions across all extended address blocks. The primary impact this has on the user is to cause the loss of the use of the last 8K section of every 64K block of physical addresses.

This modification (as shown in the schematics below) has been successfully installed on at least one board, an old MP-16 SVTFC 32K dynamic memory board previously modified for 56K operation. The new chip may either be piggybacked on an existing 16-pin chip (by straightening all legs except 8 and 16) or may be installed on the back of the board behind an existing 16-pin chip or socket (by bending legs 8 and 16 backward). The extended address for the board is established by jumpering the board side of the cut VMA line to the appropriate pin on the 74S138. Extended addressing may be temporarily disabled by connecting the 74S138 end of this jumper to pin 5 on the 74S138, which is connected to the bus side of the VMA line.

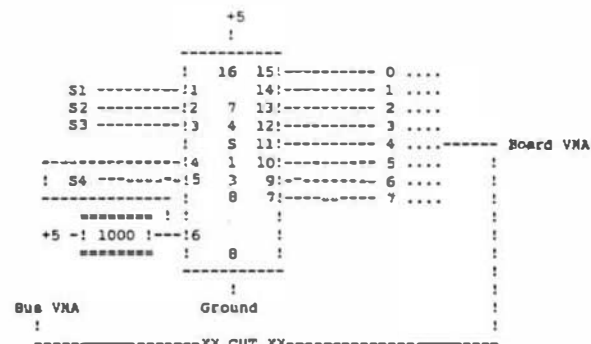
If extended addresses of 0 thru 7 are desired, the first schematic should be used. If extended addresses of 8 thru F are desired, the second schematic should be used. The extra inverter required by the first circuit may usually be found as a spare gate on the board, or may be added as a 74S04.

As shown in the third diagram below, if no DNA is present and VMA is not required by the computer configuration, the bus VMA line may be disconnected from pin 5, the bus S4 line may be moved from pin 4 to pin 5, and pin 4 may be tied to pin 6. The resulting board will decode only extended addresses 0 thru 7, but will not require the extra 74S04 inverter.

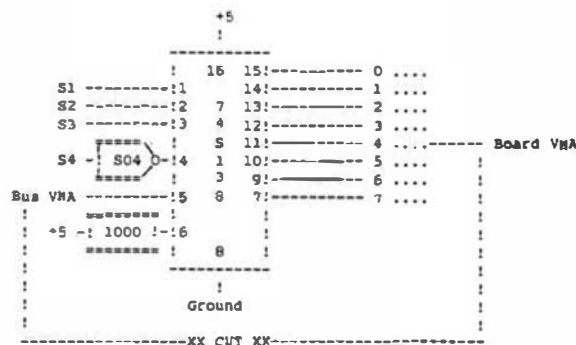
Depending upon software requirements, as little as 4K bytes (in 4K increments) or as much as 56K bytes (or more) may be required for meaningful use of the extended addressing capabilities of the machine.

The changes described here may also be used on other non-memory SS-50 boards, such as Terminus Design Arcade-50, Thomas SS-50 multiple-I/O and Video, Gimix SS-50 ACIA/PIA, Data Systems SS-50 I/O and Disk Controller and Video and other manufacturers' originally non-extended address decoded boards, to allow them to be used on SS-50C busses requiring extended address decoding.

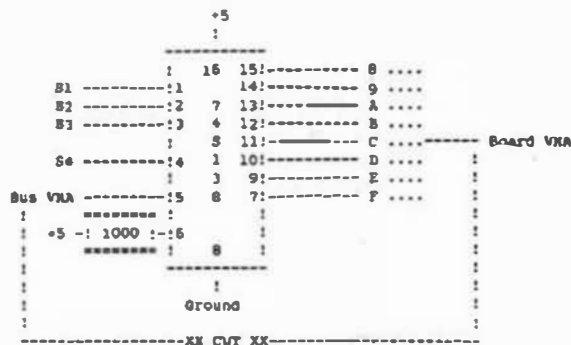
With minor adjustments, the circuits described here could also be used to correct the decoding on the CPU and mother board to recover all except one of the lost 8K blocks of physical addresses described above. In order to be compatible with most existing software, such as Flex, the CPU and I/O board functions should be decoded to extended address F. However, most software will not necessarily automatically use the additional space. VDISK, the simplest means of using extended addresses under FLEX, has an assemble-time option for using the full 64K address space in all 64K blocks (except for F000X). It treats the extended memory as a very fast disk drive.



Extended Addresses 0 Thru 7 Only - No DMA



Extended Addresses 0 Thru 7 Only



Extended Addresses 8 Thru F Only

COMPUSENSE LTD.

Computer Systems Consultants

Don Williams
68 Micro Journal.

P.O. Box 169
London N13 4HT
Tel: 01-682 0681

STRUCTURED ASSEMBLER / TRANSLATE SUBROUTINES

Dear Don,

I hope that the enclosed macros and subroutines will be of interest to you and your readers.

We have achieved programming productivity approaching that of a good high-level language by using our Structured Programming Macros (STRPMACS). The idea behind these macros is not new but the implementation we have encountered so far have been used on large mainframe computers and they made use of comprehensive assembler facilities for conditional code generation, which far exceed the capabilities of the Technical Systems Consultants 6809 Assembler for example.

We have developed what we consider to be a simple, pleasant and flexible approach to structured assembler coding, based on a set of macros designed for the TSC 6809 assembler, or an equivalent.

To overcome the limitations of the assembler, each macro call has to be related to other macro calls which originate to that logical group. This is done by tagging the calls with an identifier which is the same for all calls within one logical group. There is a bonus in that the structure of a program can be easily determined simply by looking at the related macro calls - even with the TSC editor, just by locating all the occurrences of an identifier.

The translation subroutines are modelled on certain instructions available on large IBM computers. The translate and test routine is especially useful in applications involving syntax analysis and text scanning.

The test program illustrates the use of the macros and routines. This program is in some ways trivial but took a surprising amount of time to write and test as a result of using the Structured Programming Macros and the subroutines.

If any of your readers would like a copy of these routines, we will supply them on a FLEX diskette for £5.00 (plus £1.00 for £10.00 orders sterling) which includes postage anywhere.

Yours sincerely,
Stan Opyrchal
Stan Opyrchal

© COPYRIGHT COMPUSENSE ENGLAND 1982

The following routines and macros were written in June 1982 by Stan Opyrchal and Ted Opyrchal of :-

COMPUSENSE LIMITED
P.O. BOX 169
PALMER'S GREEN
LONDON N13 4HT
Telephone 01-682-0681

Permission is granted to freely copy and redistribute this code, for non-commercial purposes, provided that these comments are included and displayed in their entirety.

COMPUSENSE STRUCTURED PROGRAMMING MACROS

For the Technical Systems Consultants 6809 Assembler.

Formats :-

IFB	N,C,L
IFNB	N,C,L
ELSEB	N,NEXT,L
IFEND	N
IFEXIT	N,L
DO	N

DOEND	N
DOEXIT	N,L
LOOP	N,L
WHILE	N,C,L
UNTIL	N,C,L
COUNT	N,I,J,L
DCOUNT	N,I,J,L
CLOOP	N,I,J,L
DCLOOP	N,I,J,L

Where :-

N is an identifier used to associate an IF or DO group. This is used to construct labels by prefixing the test supplied as N with a character as follows :-

E..... - Start of Else clause (ELSEB)
F..... - End of If group (IFEND)
D..... - Start of Do group (DO)
Z..... - End of Do group (DOEND)

C is a condition as used in 6809 conditional branch instructions. Supported conditions are :-

ED NE NI PL CS CC VS VC GT LE GE LT HI LS HS LD.

L is literally 'L' means a Long Branch is required. The assembler will tell you when this is needed. Otherwise omit. (always last parameter)

NEXT is literally 'NEXT' if no ELSE clause is to be coded in the IFB/ELSEB/ENDIF construction. This avoids the generation of a redundant branch instruction.

I,J is a specification of a preset loop count field. If a register is used as a count field then code the I parameter only. I.E. J parameter is null. For a counter in memory code I as the displacement and J as the register. J may be null for direct page locations.

Usage :-

IFB/ELSEB/IFEND/IFEXIT

This construction generates the required labels and branch instructions to control execution depending on the condition code set when the IFB is executed. The IFEXIT macro causes a branch to the location following the corresponding IFEND macro.

IFNB

This is used identically to IFB but generates the opposite logical branch.

For example :-

```
IFB ADDR,CS
Then clause
...
ELSEB ADDR
Else clause
...
ENDIF
```

The 'Then clause' is executed if the CARRY bit is set in the Condition Code. Otherwise the 'Else clause' is executed.

DO/DOEND

Define the start and end of a Do group and are used in conjunction with the following.

DOEXIT

Generates a branch to the end of the group.

i.e. The instruction following the corresponding DOEND.

LOOP

Generates a branch to the start of the Do group. i.e. The instruction following the corresponding DO.

WHILE

Generates a conditional branch to the end of the Do group if the condition is False.

UNTIL

Generates a conditional branch to the end of the Do group if the condition is True.

COUNT

Decrements the specified one byte counter and branches to the end of the Do group when the counter reaches ZERO.

DCOUNT

As for COUNT but uses a 2 byte counter.

CLOOP

Decrements the specified one byte counter and branches to the start of the Do group if the counter has NOT reached ZERO.

DCLOOP

As for CLOOP but uses a 2 byte counter.

For example :-

'68' Micro Journal

- NOW DEMONSTRATE STRONG BY TRANSLATING LOW (and) UP CASE AND PRINTING
- AFTER A FEW TIMES

```

LDA #3
DO TRAP1
  BSR TRAPAT
  CLOOP TRAP1.A
  COUNO TRAP1
*
* NOW DEMONSTRATE STRANT BY ANALYBING LINE BUFFER
*
LDX #LIMBUF LINE BUFFER
LDA #128 MAX LENGTH
DO TRT01
  LEAV DELIM,PCR DELIMITER TRANSLATE AND TEST TABLE
  LDB #07F MASK OFF TOP BIT
  BSR STRANT TRANSLATE AND TEST
  IFS TRT02,CC
*
  DELIMITER FOUND
  LEAF -1,X STEP BACK TO LOSE DELIMITER
  BSR OUTYTX PRINT TEXT BETWEEN DELIMITERS
  LEAF 1,Y RESTORE POSITION
  CMOB #TRCA IS IT CR
  UNTIL TRT01,ED FINISH IF CR
  DEC A NOTE: A IS REMAINING COUNT+1
  UNTIL TRT01,ED FINISH IF COUNT NOW ZERO
  LOOP TRT01 LOOP FOR NEXT
*
  ELSEB TRT02
*
  END OF DATA REACHED AND NO DELIMITER
  BSR OUTYTX
  IFEND TRT02
DOEND TRT01
*
JND WAKMS
*
* TRANSLATE AND PRINT CONTENTS OF LINE BUFFER
*
TRAPAT EQU *
PSHB A,B,X,Y
*
* LENGTH OF TEXT IS X - Y
  TFR X,D
  SUBD A,B LENGTH OF VALID TEXT IN BUFFER
  EFB A,B PUT RESULT ((128) IN A
  TFR Y,X START OF BUFFER IN X
  LDB #07F MASK
  LEAV UPLOW,PCR UP(==)LOW CASE TRANSLATE TABLE
  BSR STRANS DO TRANSLATE
  PULS A,B,X,Y
  BSR OUTYTX PRINT
  RTS
*
OUTYTX EQU * DISPLAY CHARACTERS STARTING AT Y ENDING AT X
*
PSHB A,B,X,Y
  TFR X,D CALCULATE LENGTH TO OUTPUT
  SUBD A,B SUBTRACT Y
  WHILE DOUT01,GT DONT BU PUT IF -VE OR ZERO
*
  JBR PCRLF START NEW LINE
DO DOUT01
  LDA 0,Y+
  JBR PUTCHR
  CMPY 2,B COMPARE WITH X VALUE ON STACK
  UNTIL DOUT01,GE
  LOOP DOUT01
DOEND DOUT01
PULS A,B,X,Y
RTS
*
LID TRANSUPS
*
* DELIMITER TRANSLATE TABLE
*
* ALL DELIMITERS TRANSLATE AS THEMSELVES
DELIM EQU *
*
  0 1 2 3 4 5 6 7 8 9 A B C D E F
FCB $00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00 - 0
FCB $00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00 - 1
FCB $20,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00 - 2
FCB $00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00 - 3
FCB $00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00 - 4
FCB $00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00 - 5
FCB $00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00 - 6
FCB $00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00 - 7
UPLOW EQU * LOW (==) CASE TRANSLATION
*
  0123456789ABCDEF
FCC / / / 0
FCC / / / 1
FCC / !-@#%&'()*+,-./:; 2
FCC / 0123456789:;<=? 3
FCC / @abcde fghijklmno 4
FCC / pqrstu vwxyz[\]^_ 5
FCC / ABCDEF GHIJKL MNO 6
FCC / PQRSTU VWXYZ[\]^_ 7
END START

```

UNIVERSAL DATA RESEARCH INC. ANNOUNCES "DISKSUB" BUBBLE MEMORY THAT LOOKS LIKE A DISK

For further information contact
Joel Heckman
Universal Data Research Inc.
2457 Wharfe Drive
Buffalo, New York 14221
716-631-3811

Universal Data Research Inc. announces "DISKSUB", a compact bubble board with 128k bytes of data storage, which will interface to the FLEX operating system using 68xx processing family, with a 30 pin as 50 i/o bus. DISKSUB acts like a disk but has the advantages of bubble memory, high reliability data storage and operation in harsh environments where no disk would dare to go. Its applications in process control, automation, data logging, and robotics are endless.

The rugged solid state bubble memory is non volatile, with no moving parts to wear, and no oxide to rub off. Also, because it is hermetically sealed, DISKSUB is unaffected by outside contaminants, high humidity, high shock, and vibration. The DISKSUB can be used to boot up systems replacing the need for disks all together.

DISKSUB is available from Universal Data Research Inc for \$995.

FLEX is a trademark of Technical Systems Consultants.

HIGH-PERFORMANCE COMPUTER PROGRAMS

ACCURATE • RELIABLE • FAST • COMPACT • MODULAR

1982 June 12

Don Williams, Sr., Publisher
68Micro Journal
5900 Cassandra Smith Rd.
P. O. Box 849
Hixson
Tennessee 37343

Dear Mr. Williams:

I have recently had the opportunity to use the Omegasoft Pascal compiler (version 2.0), and thought I'd pass along my evaluation of it.

The language Omegasoft chose to implement is excellent - it includes all of standard Pascal (with some minor exceptions which are unimportant to me and probably to most other people). It also includes significant extensions, such as the ability to access specific memory locations. I was developing a 1000+ line program, and wanted to stick as close as possible to standard Pascal. The program involves geometric and topological calculations and some fairly complicated data structures (arrays of structures which include variant records and structured fields). I believe there are only 5 lines which deviate from the standard. (They involve opening and closing files.)

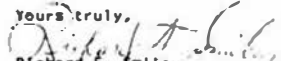
The documentation is also excellent - clear, complete, and above all accurate.

Program development is fast, because the compiler is fairly fast, and a debugger is included. On a 2 MHz S/09, compiling, loading, and running my complete program can take less than 10 minutes, including the time to edit the program slightly before compiling. Thus, I can fix several minor bugs per hour. (Compiling without output, to check program syntax, takes about 45 seconds, 10 of which is loading the compiler.)

Furthermore, most of the debugger is written in Pascal, and the source is included. I have made several slight changes in the debugger, and they were easy.

But probably the most important information for me to mention is that there are very few errors in the compiler, and support is excellent. In the course of developing this program, I have found only one situation in which the compiler generated bad code. (It involved nested records.) I called Omegasoft, and was told that bug had already been reported and a fix was in the mail. I got it the next day. After finishing the program, I sent a letter reporting some minor bugs. Nine days later I received an excellent letter responding to each of the points I had raised (and promising to fix each of the bugs in the next version of the compiler).

I am very impressed, and would like to tell Omegasoft publicly "Keep up the good work!"

Yours truly,

Richard F. Smiley
Software Engineering Consultant

27 June 1982

J. William Stull
37 Sta. Monica Dr.
Pasig, Metro Manila
Philippines

68 Micro Journal
3018 Hamill Rd.
P.O. Box 849
Hixson, TN 37343
U.S.A.

Dear Mr. Williams,

I have been a reader of 68MJ since 1979 when I built a SHWPC 4800 system using cassette storage, a PR-40 printer, and a CT-64 terminal. I later upgraded to FLEX2 and then to a 4809 cpu using FLEX9, upgrading my CT-64 to a CT-82 and my printer to a Centronix 337. My latest upgrade is to a HELIX mainframe and a GIMIX double density FIO disk controller using RUMBUG by Peter Stark as the system monitor. I made a few changes to make it all work which I would like to share with your readers who might be considering doing the same. This system has been running very smoothly ever since, even in this tropical environment.

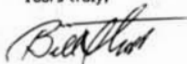
First of all, HUMBUG version 1.1, Model H094K5HTES16E is the monitor and will not work as is on the 2 Mhz HELIX system. I therefore made the following mods:
 Change F514 from \$04 to \$09 (Increase boot delay for 2 Mhz cpu)
 Change FF13 - FF19 to \$4C,\$A7,\$2E,\$A7,\$0E,\$20,\$04 (DAT handling)
 The clear screen command string at F3AD might not be correct for your terminal. For the CT-82, change the string to \$10,\$16,\$10,\$16 (repeated ROMSOP,ERASE). There is a fair amount of unused space in the HUMBUG EPROM so I added a new command to configure my terminal. It is a prompting by number type and seems to work well. It is too long to include here but I am sending it to Peter Stark.

When I went to write the data back to the EPROM, I found that the WRITPROM.CMD would not work on the 2Mhz system. I made the following changes to the program:
 Change 101E from \$2A to \$2B
 Change 1380 from \$7C to \$F9
 Change 1689 from \$32 to \$31

The WRITPROM.CMD now works very well at 2Mhz using a SWTPC MP-R programmer board. I did not have to change any parts on the MP-R Board either.

The HELIX system has been reviewed in 68M7 and in general, my comments are the same. I like to use the terminal clock to control the baud rate by software, so I added a jumper from pin 24 of the HELIX port 1 RS-232 connector to pin 15 of the header, H2 at the rear of the motherboard and cut the H2 jumper that was between pins 5 and 15. I now have a system that is very dependable and enjoyable. I have likewise enjoyed your magazine for the past 3 years. Keep up the good work!

Yours truly,



Bill Strull

GIMIX 1327 W 87th St. PL. CHICAGO, ILL. 60608 • (312) 927-5510 • TWX 810-221-4050

Press Release

INTELLIGENT SERIAL I/O PROCESSOR BOARD NOW AVAILABLE

The GIMIX Intelligent Three-port RS-232C Serial I/O Interface can significantly increase throughput of a multi-user system by reducing the number of interrupts between user terminals and the host CPU. The Intelligent I/O Board accomplishes this by buffering data transfers between system and users and preprocessing of the data.

Appropriate on-board software and operating system drivers are required. Software and drivers for OS-9 Level 2 will be available shortly from GIMIX.

*INDEPENDENT ON-BOARD 2MHz 68809 CPU

*UP TO 20K OF ON-BOARD MEMORY (EPROM AND RAM)

*BUFFERED DATA TRANSFER BETWEEN HOST AND ON-BOARD CPU USING A 28038 FIO I/O INTERFACE UNIT

*THREE RS-232C SERIAL I/O PORTS (65514s) WITH SOFTWARE SELECTABLE BAUD RATES, WORD LENGTH, STOP BITS, PARITY

STANDARD VERSION INCLUDING 64K RAM WITHOUT SOFTWARE\$439.00
 PARALLEL VERSION COMING SOON

UNIFLEX FOR GIMIX WINCHESTER SYSTEMS

TSC will be providing UNIFLEX compatible with GIMIX Winchester systems. The NEW versions of UNIFLEX for use with the Winchester systems will be delivered on 5" media as well as 8" media.

GIMIX 30 PIN PROTOTYPING BOARD NOW AVAILABLE

*Double sided with plated thru holes and gridded power and ground lines.

*8 rows of pads on .100 x .300 centers: up to 41 fourteen pin ICs.

*Accepts standard 6, 8, 14, 16, 20, 24, 28, and 40 pin DIP devices.

*The entire top edge has pads for .100 x .100 header (ribbon) connectors.

*Pads for solder connections or .100 center headers on all 30 bus lines.

*Accepts 3 70-320 regulators, 1 on the +5V & 1 on the +/- 16V lines.

*Provisions for decoupling caps distributed throughout the array.

*Can be used with wire wrap, wiring pencil, solder wiring, etc.

With gold bus connectors and heat sinks -- unassembled \$58.33

Mr. Don Williams
 '68 Micro Journal
 P.O. Box 849
 Hixson, Tenn. 37343

June 26 '82

Dear Don;

I was very pleased with your note for a better EPROM programmer which is available from SWTPC. (I can not recall the exact page or issue of 68 micro.). I took another approach to have a programmer which can work at 2 Mhz system clock, patching WRITPROM.CMD. I not only faced the inconvenience of 1 Mhz clock use in WRITPROM.CMD but also the possible damaging of the some 2716's from differnt sources. The 2716's from Intel and 2516 from Texas Instruments are programed fine with WRITPROM.CMD but the 2716's from Natinal Semiconductors and AMD are damaged. 50 m.sec. PG/PG4 program pulse seems to be too long for these 2716's. To solve thease problems I tried a few patches at WRITPROM.CMD.

Using DYNAMITE.CMD disassembler I could locate the basic time loop routine (1 m.sec. at 1 Mhz clock) at \$137E,

```
137E 8E 007C L137E LDX #007C 1 m.sec. loop
1381 30 1F L1381 LEAX -1,X
1383 26 FC BNE L1381
1385 5A DECB 1X(ACCB) m.sec.loop
1386 26 F6 BNE L137E
1388 39 RTS
```

50 m.sec. program timing is located at \$1282,

```
1282 C6 32 LDB #32 50 m.sec.
1284 BD 137E JSR L137E gosub timing loop
```

Also I noticed the program checks CPUTYP clock frequency at \$1018,

```
1018 7D CC33 TST CPUTYP
101E 2A 08 8PL L1028 if 1 Mhz goto next step
1020 8E 1664 LDX #1664 get 2 Mhz notice message
```

Now it is obvious that \$7C at \$1380 to be changed

to \$FA and \$2A at \$101E to \$28 (8MI instruction) for 2 Mhz clock, and \$32 at \$1283 to be changed to \$25 for 37 m.sec. PG/PGM program timing interval. Patches for each modification are shown in following table.

	WPROM1TI	WPROM1NS	WPROM2TI	WPROM2NS
Clock	1 Mhz	1 Mhz	2 Mhz	2 Mhz
Supplier	TI,Int	NS,AMD	TI,Int	NS,AMD
\$101E	\$2A	\$2A	\$28	\$28
\$1283	\$32	\$25	\$32	\$25
\$1380	\$7C	\$7C	\$FA	\$FA
\$1689	\$32	\$32	\$31	\$31

TI: Texas Instruments
 NS: National Semiconductors
 Intl: Intel
 AMD: Advanced Micro Devices

The READPROM.CMD works at 2 Mhz if 68820 is used at MP-R programmer board. I found a 6820 which works properly at 2 Mhz, so you may try too.

Now I can use cheapest 2716-1 (2 Mhz) for programing. I spent a whole Friday night for above but it is my hobby to do so. So I should not feel guilty not to read any articles or books for my own profession, nor stupid not to have a good time at one of the restaurants near by.

Sincerely yours;



Hiromasa Kitazume, R.D.
 11720 Edgewater Dr.
 Lakewood, Ohio 44107

SOFTWARE EPROM SELECTION FOR THE SCB-69

Richard H. Roe, CBT
 Unique Technologies
 P.O. Box 671
 Exporia, VA 23847

Shore Signal Broadcasting's SCB-69 CPU board is an extremely versatile, quality product for the Fifty bands. However, those of us desiring to run both D0669 and OS-9 on this board have had to tolerate swapping chips, because both operating systems require an EPROM residing at \$F800 to \$FFFF. This article describes a relatively simple modification which eliminates the problem.

The SCB-69 provides five sockets for 2K EPROMs. In the standard configuration the highest EPROM resides at \$F800 to \$FFFF, with the other four sockets situated in a block of memory from \$D000 to \$EFFF. In a 56K system, the two lowest sockets are overlaid by RAM and are therefore useless. This modification moves these sockets to the same address as the two highest EPROMs and allows the user to select, via software, which pair will be accessed. Perhaps the most interesting aspect of this conversion is that Shore Signal was kind enough to provide us with almost all the logic needed for the conversion, requiring us to supply only a quad NAND package and its socket!

Begin by examining the SCB-69 CPU board carefully. Locate sockets U21 through U25 and wiring blocks W4 through W8 located directly above them. Cut jumper 4 on all but W6. On W7 and W8 cut jumper 2 also.

You are now going to make a few jumper connections to the wiring blocks. Looking at the front (component side) of the board, notice that the LEFT vertical row of pads for each wiring block is numbered 1 through 4. All connections should be made to the left or numbered side of the wiring blocks. Do not make any connections to the pads on the right side.

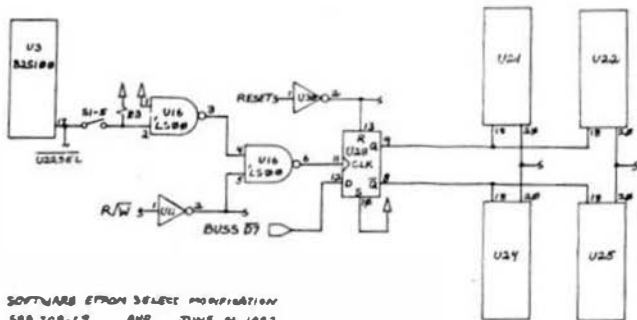
Jumper 4 on W4 to 4 on W5; also jumper 4 on W7 to 4 on W8.

Jumper 2 on W6 to 2 on W7; also jumper 2 on W5 to 2 on W6.

Install a 14 pin socket in the spare location U16. Insert the socket so the lower two pins of the 16 pin PC pattern are unused. Solder a jumper from pin 7 of the 14 pin socket to W6, which runs immediately below the socket.

Now a list of corrections to make:

U16 PIN 1 TO U16 PIN 14
U16 PIN 2 TO S1 PIN 5
U16 PIN 3 TO U16 PIN 4
U16 PIN 5 TO U11 PIN 2
U16 PIN 6 TO U28 PIN 11
U28 PIN 8 TO W7 PIN 4
U28 PIN 9 TO W4 PIN 4
U28 PIN 10 TO U28 PIN 14
U28 PIN 12 TO S50 PIN 43 (D7)
U28 PIN 13 TO U30 PIN 2
S1 PIN 6 TO U3 PIN 17
S1 PIN 5 TO U3 PIN 6



SOFTWARE EPROM SELECT MODIFICATION
580 SCB-69 AHR JUNE 4, 1982

On the connections to W4 and W7, remember to connect to the left side of the blocks, looking at the front of the board. Pin 43 on the S50 buss is the eighth pin from the right side, on the component side of the board. Pin 5 of S1 is next to pin 6 of S3; you should be able to just bend these pins together and solder.

Install a 74LS00 in U16 (a 7400 will work as well.) Your EPROMs go in as follows: The program to be executed on power-up or on reset goes in U21 (I prefer ROM69 here). If you have a companion chip it goes in U22. Install your secondary EPROM(s) in U24 (and U25). Remember that, when switched, U24 corresponds to U21 and U25 replaces U22. Any EPROM placed in U23 is active at all times. Here is a typical setup (mine): ROM690 in U21, FLEX900 in U22, nothing in U23 (yet!), OS-9 P1 in U24, and OS-9 P2 in U25.

Now to set the switches. S1-5 controls the software select option. In the off position, U21 through U23 are always active; U24 and U25 don't exist. In the on position software select is enabled. S2-1 enables U21 or U24 and MUST be on. If you have an EPROM in either U22 or U25, you must also turn on S2-2. S2-3 controls U23 as always. S2-4 and S2-5 now have no function and should be left in the off position. S2-6 through S2-8 operate as before and should be left alone.

Turning power on or hitting reset will always select the U21-U22 pair of sockets. To select the U24-U25 pair, write any number with bit 7 off (00 is easy to remember) to any location between \$E800 and \$EFFF. This is an EPROM select - either U22 or U25 - and as such is never written to under normal conditions. To select the U21-U22 pair again, write any number with bit 7 on (I use \$FF) to the same range of locations. Hitting RESET also accomplishes this. Thus my system comes up in ROM690. To boot ROM690 I hit "0"; "8" boots FLEX900, and running the following program boots OS-9:

```
COLDFLT EQU $FFAE RESET VECTOR IN MY COPY OF OS-9
ROMCTL EQU $E800 ADDRESS WHICH CONTROLS EPROMS
GO OFF LDA #00 GET AUXILIARY EPROM SELECT CODE
STA ROMCTL SELECT OS-9 EPROMS
JMP COLDFLT COLDFLT: OS-9
```

Similarly, substituting the ROM690 or FLEX boot vector and SPP for the select code gets me back from OS-9 to DOS or FLEX. Variations of these programs work well as transient commands; with proper care one can move from DOS to FLEX to OS-9 in any sequence, at will.

I am certain there are ways to do this which are much more flexible, but for about fifty cents in parts, a half hour's time, and flawless performance this simple modification can't be beat.

WARNING: This modification will most certainly void any warranty you may have on your CPU board! This modification was installed on REV 3 of the SCB-69 CPU board; others may be different. If you are using one of the alternate EPROM memory maps shown in the SCB-69 manual you should be sure that no conflicts will occur before attempting this modification.

FLEX is a trademark of Technical Systems Consultants, Inc.
OS-9 is a trademark of Microware Systems Corporation.
SCB-69, ROM69 and ROM9 are trademarks of Smoke Signal Broadcasting.

378 Broadbrook Ave.
Minneapolis, Minnesota

June 2, 1982

Don Williams
C/O 68 Micro Journal

Dear Don,

Having recently assembled Digital Research Computer's 32K SS-30 still, you sit I recommend it to those considering such a kit. The 32K kit board is an excellent quality, double sided board with plated through holes. It is color coded and silk screened with all

component locations clearly marked. The documentation is adequate for someone with some experience assembling kits but is not for the neophyte. Assembly instructions are of the form: (install) disk controllers at locations C1 to C10. The board is contained to be fully populated and must be addressed on a 32K boundary. As none of the memory can be disabled you can only address the board on an odd 32K boundary if extended addressing has been selected. My board has performed flawlessly in my 6800 system since I assembled it.

If you live in Canada be prepared for a long wait (the kit is shipped via surface mail), two weeks to process the order plus four weeks for shipping. My kit was shipped with the wrong socket set which added another two week delay. My request for the correct socket set was handled with reasonable courtesy and promptness.

The kit appears to be a good product worth purchasing. Now that I have read seven issues of your publication and many purchases from your advertisers (some of whom I never knew existed before), I feel that your magazine is worthwhile. I hope to see more product reviews and technical information.

Sincerely,
Randy Cooper
Randy Cooper

Electronic Component Service, Inc.

212-846-2600

89-06 130TH STREET
RICHMOND HILL, NY 11418

June 29, 1982

Robert May
48 Micro Journal
5900 Cassandra Smith
Riverton, TN 37343

Dear Bob,

Here is a report on my experience with Westchester Applied Business Systems Software (WABS): The Data Management System, and Basic Accounting System. WABS software was designed to run under file and any 6809 CPU. His ad caught my eye in 68 Micro Journal so I sent for the user guides. After thoroughly examining them, a demo would be in order. I met with Bill Adams of WABS and he answered all my questions besides giving me an excellent demo of the software capabilities. I purchased the system on the spot.

The following are my findings after using WABS's software.

Basic Accounting System

The basic accounting system filled all the needs of my business plus some. The great thing about the WABS accounting package is everything is included. No need to buy separate packages. It already contains a provision for a products and accounts file and provides the following reports: Journal, ledger, balance, income, and monthly class program.

To set up the entire accounting system took me approximately 1 hour. It took me that long because I never worked with a computer before. The instructions included in the user guide are in simple English with excellent examples. Keep in mind that the accounting system is generic. The user defines to the system the size and format of the accounts and products file. Also I neglected to mention that the accounting package includes accounts payable and receivable.

Data Management System (DMS)

The DMS consists of the following subprograms: The DMS Run, Define, update, Generate, format, and WABS programs. Taking them one at a time are as follows. The DMS Run subprogram simply contains over 200 common functions used by WABS.

The Define program enables the user to specify the format and content of a database file. I have in the past year 3 files to track various items dealing with customer accounts and products. Defined.

The update program lets the user edit any of the databases. The user is able to insert, review, change, and delete database records. To use the update program the database needs to have been previously defined by the user.

The most versatile program of the DMS is Genar. To sum it up in a word would be magic. Genar does English like non procedural language to produce reports, inquiry, and to generate new databases. Genar can be used to sort databases, combine information from different databases and calculate any specified field in any direction. Genar will take output fields from different databases and group them together in any desired format, run calculations on them, reformat them and build new files containing that information. Genar is useful in creating control files for batch processing. The list of Genar commands totals approximately 37.

Next in line is the Format Program. I can't say too much about it because I've never used it. Recently I received a bill from WABS; I found later it was printed by the format program. So I contacted WABS as to how to do this with my system. I was surprised to find that I already had the same capability.

The final program I will talk about is WGEN. It is used to format a paginated disk to expand the computer memory to over a megabyte.

My personal opinion of WABS's software is first class all the way. It has served me well in business and I certainly got my money's worth out of it. I would recommend it to anyone looking for good solid application software system at a reasonable price. Bill Adams of WABS has been very helpful to me and I want to take this opportunity to thank him for his patience in helping me make the best software decision that I possibly could.

Sincerely,
Randy Cooper
Randy E. Strecher

Don Williams, Sr., Editor
68 Micro Journal
5900 Cassandra Smith
Computer Publishing Center
PO Box 649
Riverton, TN 37343

RE: ENLARGING THE FLEX DIRECTORY

The following discussion applies specifically to Flex formatted double sided, double density (not extra density) diskettes. However, similar logic can be applied to any Flex diskette. Normally, a Flex diskette starts out with a directory on track 00. This directory starts with sector 05 and continues thru sector 3E, yielding 26 sectors, each containing 10 entries. For a total of 260 files. If more than 260 files are needed in the directory, Flex will automatically extend the directory. However, Flex will only extend the directory one sector at a time, yielding only 10 additional directory entries. Ten files later, Flex will once again have to extend the directory. If a system disk contains 361 files, then Flex will have had to extend the directory 11 times. Flex users know that this creates a significant

delay in file access. What follows is a crude but effective method of providing room for 780 file entries before Flex will have to extend the directory. And all it costs the user is one track which is adjacent to the original directory!!!

Using the BUMPREP (DRI utility) as published in 68 Micro Journal, do the following to a newly formatted diskette in drive A1....

```

      CHANGE
CMD DRIVE TNO SCL BYTE FROM TO ACTION
-----
R      01 00 03      10 01 02 Select System Info Rcrd
<CR>
R      22 70 3C Cno from 3952 to 3900 free sectors
<CR>
M      Write modified SBR
<ESC>
R      01 00 1E      00 00 01 Select last sector of old directory
      01 00 01 Modify last sector to link to 01-01
      (extending directory 52 sectors)
<CR>
M      Write modified old last sector
<ESC>
R      01 01 34      00 02 00 Select last sector of track 01
      01 01 00 Break link to 02-01
<CR>
M      Write 1st sector of extended directory
      Return to Mother Flex

```

Of course a more elegant method would be to disassemble the NEWDISK command and have it do this reformatting at the end of its initialization, saving head to the type of diskette of course. Or you could write a simple utility using Flex's single sector read and write vectors. But the above method was thought out in a few minutes and then the following procedure is now observed....

- (1) Format new diskette with UDF or NEWDISK
- (2) Mirror a diskette containing the extended directory onto the newly formatted diskette. Now it has an extended directory too!

Best Regards,

Bill Fisher

Mr. Don Williams
68 Micro Journal
5900 Cassandria Smith
P.O. Box 849
Hixson, TN 37343

5469 Ariene Way
Livermore, California 94550
(415) 455-6085
June 17, 1982

Dear Don:

I am pleased to see more and more articles in 68 Micro on the "C" programming language. However, the only compilers available seem to generate code for the 6809. For those of us who still use the 6800, I now have a solution. Over the past year or so, I have converted Ron Cain's Small-C compiler to generate a generalized type of pseudo-code. A run-time interpreter executes this pseudo-code at a reasonable rate. Alas, the 6800 instruction set is not suited for direct code generation by the compiler, at least not if you want to fit the compiler in under 32K.

If you or any of your contributing editors are interested, I will be happy to furnish the source code for the compiler, interpreter and the run-time library for publication in your magazine. This compiler was originally written for the 8080 and placed in the public domain by Ron Cain via Dr. Dobbs Journal, and I would like to see the same happen for the 6800 community. I personally plan to get filthy rich by charging \$29.95 per copy of the above on an 8 inch Flex diskette.

Sincerely yours,

Serge Stepanoff
Serge Stepanoff

Joseph DeHoff
R.R. 4
Centerville, IL 62801
618-533-5270
July 9, 1982

68 MICRO JOURNAL
PO Box 849
Hixson, TN 37343

Dear Don,

In BIT BUCKET of the July issue Keith Alexander asked for a better algorithm to find perfect numbers. I hope the enclosed program will help him. It only finds the first 7 perfect numbers as the 8th is over 16 digits long; by the way, the 10th is over 32 digits long and will give an overflow even on the IBM 370 using PL/I.

Euclid found that any number of the form $(2^n - 1) * 2^{(n-1)}$ was perfect if $(2^n - 1)$ was a prime, so only numbers of this form are tested. The first 7 are found in less than 20 seconds.

Anyone for writing a BASIC program to find the first 10??

Thank You,

Joseph DeHoff
Joseph DeHoff

```

15 PRINT*      P1      TIMES      P2      =      A PERFECT NUMBER
20 PF=2
25 PORT=1707
30 P2=PF
31 PF=PF*2:P1=PF-1
32 IF P1=1 GOTO 70
40 PORT=3707:(P1)STEP2
50 IF INT(P1/TL)*TL=P1 THEN TL=P1
55 NEXT TL
60 IF TL>P1 THEN GOTO 30
70 PP=P1*P2:PRINT P1,P2,PP
80 NEXT P1
90 END

```

P1	TIMES	P2	=	A PERFECT NUMBER
3	2			6
7	4			28
31	16			496
127	64			8,128
8,191	4,096			33,550,336
131,071	65,536			8,589,869,056
524,287	262,144			137,438,691,328

CLASSIFIED ADVERTISING

6800 CPU Card \$75. AC-30 Cassette Interface with documentation \$50.
LEW 1-615-842-4601

Tano 56K Dual Serial Ports, Dual 5" Disk Drives, complete software package. Like New, Wholesale \$5300 - will take \$2995. Three SWTPC 8K Memory Boards \$95 ea. PR-40 Printer, Extra Paper, and Ribbons; \$95.
LEW 1-615-842-4601

DSO color graphics board \$50, PR-40 printer \$50, both working and socketed.
Eric Skarjune work (313) 232-8181 ext 265 home 732-1917.

SWTPC MF-68 Dual Mini-Floppy, \$500; MP-16A, MP-16X, PR-40, \$150 each; MP-A2, \$60; MP-A, Universal Bare Motherboard, \$40 each; MP-M 4K (5), MP-R, Newtech Music Board, \$30 each; MP-S (3), MP-C, MP-LA, AC-30, CIS-30+, \$25 each; Elektra cabinet w/power supply, \$300; CREI 6802 Microcomputer, \$200; 35 issues 68 Micro Journal, \$35.
Dick Efnor, 4446 Chase Park, Annandale, VA 22003 (202) 256-1197.

FOR SALE: BFO 5 inch disk controller with SWTPC 6809 CPU. Runs Flex or SSB at \$8000 or \$6000. APF Imagination 6800 color computer with ROM Basic, joysticks and Basic Tutor. Other 6800/09 boards.
Call John (216) 747-6781 anytime.

SWTPC DMAF-2 DMA Disk Controller Board, rev C Up to 4 DSDD 8" drives. \$365.
Dave (312) 382-5478 after 6.

MICRO-CHROMA 68 with SS-50 Interface \$250. Digital Research 32K RAM 16K of chips \$100. S.E. MICRO SYS. DDC-16 \$150. TI S-R/R 48K w/8K of chips \$100.
A. Stahley (215) 261-0176 after 5:30pm EST.

FOR SALE: SWTPC AC-30 cassette interface complete with audio cables. And the original MP-A1 6800 processor board (purchased in October 1976; a real antique!) Both in mint working condition. The first reasonable offer takes either or both.
Dwight Lanpher, Box 472, N.E. Harbor, ME 04662

GIMIX PIO 28 Disk Controller Card. Will Control Up to four 5 1/2 inch drives single or double density 35 to 80 tracks, one or two sided. Includes Gimix flex disk. List \$390 asking \$275.
Call (414) 332-3242 nights (414) 462-3700 days Bob.

SWTP 6800 System, 48K, MP-S, AC-30, \$850; SSB Triple Mini-Floppy \$850; BANTEM CRT \$600.
Joe Wicklund, 801 Duchess Rd, Bethel Wa 98011, (206) 481-3573 Eve. , 347-1292 Days.

DISK DRIVE WOES?
PRINTER INTERACTION?
MEMORY LOSS?
ERRATIC OPERATION?

Don't Blame The Software!

Power Line Spike, Surges & Hash could be the culprit! Floppies, printers, memory & processor often interact! Our patented ISOLATORS eliminate equipment interaction AND curb damaging Power Line Spikes, Surges and Hash. MONEY BACK GUARANTEE!

- ISOLATOR (ISO-1) 3 filter isolated 3-prong sockets; Integral Surge/Spike Suppressor; 1875 W Maximum load, 1 KW load any socket \$69.95
- ISOLATOR (ISO-2) 2 filter isolated 3-prong socket banks; (6 sockets total); Integral Spike/Surge Suppression; 1875 W Max load, 1 KW either bank \$69.95
- SUPER ISOLATOR (ISO-3) similar to ISO-1 except double isolation & suppression \$104.95
- SUPER ISOLATOR (ISO-11) similar to ISO-2 except double isolation & suppression \$104.95
- MAGNUM ISOLATOR (ISO-17) 4 Quad isolated sockets; For ULTRA-SENSITIVE Systems \$181.95
- CIRCUIT BREAKER, any model (Add-CB) Add \$9.00
- REMOTE SWI CH, any model (Add-RS) Add \$18.00

AT YOUR
DEALERS

MasterCard, Visa, American Express
ORDER TOLL FREE 1-800-225-4876
(except AK, HI, PR & Canada)

Electronic Specialists, Inc.
171 South Main Street, Box 389, Natick, Mass. 01760
(617) 655-1532



LUCIDATA PASCAL

If YOU need an easy-to-use, well proven implementation of Pascal, that doesn't need a mini computer to run it....
LOOK NO FURTHER - WE SELL IT!

- ONLY requires 16K.8K RAM plus one 5" disk drive
- All standard Pascal types are supported and full type-checking is performed
- SEALY fast 9-digit precision scientific functions
- Optional allocation of variables to absolute memory locations allows easy control of memory-mapped peripherals (eg video boards, PIA's A/D's etc)
- Fully optimized run-time systems for 6800 and 6809
- Any number of EXTERNAL user supplied routines may be easily linked to Pascal defined function and procedure identifiers within your program
- Fast, easy-to-use compiler generates ultra efficient position independent and ROMable P-code instructions
- Multiple files - may be sequential/random disk files supported by your DOS, or physical devices added by you
- Scalar I/O to READ and WRITE your own SCALAR TYPES
- Custom versions available to special requirements
- All LUCIDATA Pascals are supplied with a comprehensive User Manual PLUS lots of demo programs and are fully supported by our Update Service

6800 FLEX 2 version on 5" disk \$150, on 8" disk \$165
6809 FLEX 9 version on 5" disk \$190, on 8" disk \$205
User Manual separately \$25 (deductible from purchase)
Prices include airmail postage ANYWHERE in the WORLD!
Payment by Mastercard/VISA or certified cheque
Despatched by return of post (UK → East Coast 5 days)



LUCIDATA Ltd., P.O. Box 128
Cambridge, CB2 5EZ, ENGLAND
Telephone Orders (0223) 841906

Purveyors of Pascal since 1979

LUCIDATA Pascal is also available for Heath HDOS from POLYBYTES, 325 19th Street SE., Cedar Rapids, IA 52403
& Smoke Signal Broadcasting DOS68D from WINDRUSH Micro Designs, Gaymers Way, North Walsham, NR26 0AN, ENGLAND

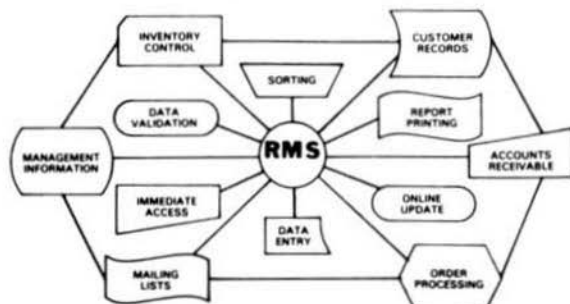
6809

RECORD MANAGEMENT SYSTEM

RMS

DATABASE MANAGEMENT

- USER DEFINED RECORD FORMAT VIA DATA DICTIONARY
- SCREEN ORIENTED, FORM FILL OUT TYPE OF ACCESS
- OPTIONAL TWO LEVEL RECORD HIERARCHY
- ALL FILES IN ASCII TEXT FORMAT, BASIC COMPATIBLE
- DIRECT ACCESS BY KEY FIELD, MULTIPLE INDEX FILES
- EXTENSIVE DOCUMENTATION, SAMPLE APPLICATION
- VERSATILE, PROFESSIONAL QUALITY REPORT WRITER
- BUILT-IN SORT/MERGE
- EASY TO USE



RMS is a complete DATABASE MANAGEMENT package for the 6809 computer. It is made up of five machine language programs that make up the most powerful business programming tool available for the 6809. It can be used by the relative novice, to implement an incredible variety of information storage and retrieval applications, without any programming. However, the programmer can use RMS as part of the solution to a larger problem, saving many hours of unnecessary program development time. RMS can be used to handle data input, editing, validation, on-line retrieval, sorting and printed reports. Custom data manipulation can be filled in by the user's BASIC programs.

SINGLE CPU LICENSE

FLEX*	\$200
OS-9+	\$250
UNIFLEX*	\$300

TERMS: VISA / MC / PREPAID

**WASHINGTON
COMPUTER SERVICES**
3028 SILVERN LANE
BELLINGHAM, WA 98226
1 (206) 734-8248

In Australia & Southeast Asia, Available Through:
Pan's Radio Electronics, 7A Burton St.
Darlinghurst, NSW 2010 Sydney Australia

* FLEX and UNIFLEX are trademarks of Technical System Consultants Inc.; + OS-9 is a trademark of Microware



DATA SYSTEMS "68"
ORLANDO, FLORIDA

5 1/4" DOUBLE DENSITY DISK CONTROLLER

- Double sided, single/double density
- Write-precompensation
- Regulators to power 2 disk drives
- Disk drivers included
- Phase lock loop data separator
- Uses the Western Digital 1791 controller chip
- Requires oscilloscope and voltmeter for setup

\$39⁵⁰ bare board

\$225⁰⁰ assembled and tested

6809 CPU BOARD

\$32⁵⁰

- On-board Baud Rate Generator using the MC14411
- Supports extended addressing
- Space for four 2716 Eproms allowing you to switch between sets
- Solder masked top and bottom
- Component placement silk screened on top of board

64K DYNAMIC RAM BOARD

\$39⁵⁰

- Completely transparent refresh (during 01) at 1 MHz
- Operates with both 6800 and 6809 systems
- Compatible with the 20-bit extended addressing mode
- Low power - 12V at 150 ma., 5V at 500 ma., and - 5V at 7 ma.
- Uses 4116-type RAM with 200 NS access time
- Designed for Motorola MC3242A and MC3480 Dynamic Memory Control Chips
- No timing problems
- No one-shot delays
- No adjustments

MOTHER BOARD

\$62⁵⁰

- .093 board no flex
- 8 50-pin slots 8 30-pin slots
- 4 or 16 addresses per slot
- fully decoded
- I/O configuration by way of 1 16-pin header and 4-pos dip switch
- Baud Rate Generator is on board 9600, 4800, 600, 300 110

Data Systems "68"
2316 Diversified Way
Orlando, Florida 32804

(305) 425-6800

DUAL SERIAL BOARD

\$19⁵⁰

- Combines two ACIA's at the same port
- An optional NMI debouncing circuitry is on board
- All line are RS-232 levels

6845 VIDEO DISPLAY BOARD

\$32⁵⁰

- Utilizes the 6845 CRT Controller
- Software selectable format up to 80 by 24
- Character font in one 2716 Eprom
- 2K Screen Buffer
- Selectable on any odd 2K boundary

MULTIPLE I/O BOARD

\$32⁵⁰

- 2-ACIA's and 2-PIA's
- Regulated + 12, - 12, and + 5 available for key board
- Addressable from \$E7EX-\$E7FX
- Addressed to overlay Motherboard I/O
- Buffered handshake on Serial I/O

MODEM BOARD

\$19⁵⁰

- Uses the MC6860 Modem Chip
- Discrete active filters
- Works with a CBT Data Coupler
- Bell 103 compatible
- Originate and auto answer
- With the MC6850 it looks like a serial port to software
- No hard-to-get parts

- All Boards Solder Masked Both Sides
- All Silk Screened Nomenclature
- Full Documentation Included
- VISA & Master Card Accepted
- Add \$3⁰⁰ for shipping & handling

TERMS: Check, MasterCard, Visa. Sorry, NO C.O.D.'s. Florida residents add 5% sales tax. Specify type board(s) and quantity of each when ordering.

TEN MOST-ASKED QUESTIONS ABOUT **DYNACALC™** THE ELECTRONIC SPREAD-SHEET FOR 6809 COMPUTERS

1. **What is an electronic spread-sheet, anyway?**
Business people use spread-sheets to organize columns and rows of figures. DYNACALC simulates the operation of a spread-sheet without the mess of paper and pencil. Of course, corrections and changes are a snap. Changing any entered value causes the whole spread-sheet to be re-calculated based on the new constants. This means that you can play, 'what if?' to your heart's content.
2. **Is DYNACALC just for accountants, then?**
Not at all. DYNACALC can be used for just about any type of job. Not only numbers, but alphanumeric messages can be handled. Engineers and other technical users will love DYNACALC's sixteen-digit math and built-in scientific functions. There's even a built-in sort command, so you could use DYNACALC to manage small data bases - up to 256 records.
3. **What will DYNACALC do for ME?**
That's a good question. Basically the answer is that DYNACALC will let your computer do just about anything you can imagine. Ask your friends who have VisiCalc, or a similar program, just how useful an electronic spread-sheet program can be for all types of household, business, engineering, and scientific applications.
4. **Do I have to learn computer programming?**
NO! DYNACALC is designed to be used by non-programmers, but even a Ph.D. in Computer Science can understand it. Built-in HELP messages are provided for quick reference to operating instructions.
5. **Do I have to modify my system to use DYNACALC?**
Nope. DYNACALC uses any standard 6809 configuration, so you don't have to spend money on another CPU board or waste time learning another operating system.
6. **Will DYNACALC read my existing data files?**
You bet! DYNACALC has a beautifully simple method of reading and writing data files, so you can communicate both ways with other programs on your system, such as the Text Editor, Text Processor, Sort/Merge, RMS data base system, or other programs written in BASIC, C, PASCAL, FORTRAN, and so on.
7. **How fast is DYNACALC?**
Very. Except for a few seldom-used commands, DYNACALC is memory-resident, so there is little disk I/O to slow things down. The whole data array (worksheet) is in memory, so access to any point is instantaneous. DYNACALC is 100% 6809 machine code for blistering speed.
8. **Is there a version of DYNACALC for MY system?**
Probably. You need a 6809 computer (32k minimum) with FLEX or UniFLEX operating system. A version for OS-9 is also in the works. You also need a decent CRT terminal, one with at least 80 characters per line, and direct cursor addressing. If your terminal isn't smart enough for DYNACALC, you probably need a new one anyway. The UniFLEX version of DYNACALC also allows you to mix different brands of terminal on the same system. There's also a special version of DYNACALC for Color Computers equipped with FLEX and DataComp's F-MATE. A version for Frank Hogg's Color Computer FLEX is also being done.
9. **How much does DYNACALC cost?**
The FLEX versions are just \$200 per copy; UniFLEX version \$395. Foreign orders add \$10 per copy for postage. We encourage dealers to handle DYNACALC, since it's a product that sells instantly upon demonstration. Call or write on your company letterhead for more information.
10. **Where do I order DYNACALC?**
See your local DYNACALC dealer, or order directly from CSC at the address below. We accept telephone orders from 10 a.m. to 6 p.m., Monday through Friday. Call us at 314-576-5020. Your VISA or MasterCard is welcome. Please specify diskette size for FLEX versions. Software serial number is required for the UniFLEX version of DYNACALC.

ORDER YOUR **DYNACALC™** TODAY

Foreign Dealers:

Australia & Southeast Asia: order from Paris Radio Electronics, 7A Burton St., Darlinghurst, NSW 2010 Sydney. Telephone: 02-357-5111.

United Kingdom: order from CompuSense, Ltd., PO Box 169, London N13 4HT. Telephone: 01-882-0681.

Scandinavia: order from Swedish Electronics hk AB, Murargatan 23-25, Uppsala S-754 37 Sweden. Telephone: 16-25-30-00.



Computer Systems Center
13481 Olive Blvd.
Chesterfield, MO 63017
(314) 576-5020

UniFLEX software prices include maintenance for the first year.

DYNACALC, DYNAMITE, and DYNAMITE +
are trademarks of Computer Systems Center.

F-MATE is a trademark of DataComp.
VisiCalc is a trademark of VisiCorp.
OS-9 is a trademark of Microware and Motorola.
FLEX and UniFLEX are trademarks of TSC.

ALSO FROM CSC

DYNAMITE + "THE CODE BUSTER"

now available for UniFLEX
OS-9 version soon

DYNAMITE + is a new version of DYNAMITE, our popular 6809/6800 disassembler package for 6809 FLEX. Present users of DYNAMITE can upgrade to DYNAMITE + by sending us the original DYNAMITE diskette and \$40 (plus \$5 for foreign postage). DYNAMITE + does everything DYNAMITE does, and more! A cross-reference generator has been added, label files are now maintained only in text form (LABEL EQU \$xxxx), and boundary file specifications have been tremendously simplified, which makes it easier to disassemble large programs containing lots of big tables.

The UniFLEX version of DYNAMITE + does everything the FLEX version does, and also automatically handles system calls and 'info' areas.

DYNAMITE + is available for \$100 per copy on FLEX (specify diskette size), and \$300 on UniFLEX. Foreign orders add \$5 per copy for postage.

68 MICRO JOURNAL PROGRAMS on DISK

Disk #1: FILESORT, MINICAT, MINICOPY, MINIFMS, **LIFETIME, **POETRY, **FOODLIST, **DIET.
Disk #2: DISKEDIT w/ Inst. & fixes, PRIME, *PMOD, **SNOOPY, **FOOTBALL, **EXPAN, **LIFETIME.
Disk #3: CBUG09, SEC1, SEC2, FIND, TABLE2, INTEXT, DISK-EXP, *DISKSAVE.
Disk #4: MAILING PROGRAM, *FINODAT, *CHANGE, *TESTDISK.
Disk #5: *DISKFIX 1, *DISKFIX 2, **LETTER, **LOVESIGN, **BLACKJAK, **BOWLING.
Disk #6: *PURCHASE ORDER, INDEX (Disk file indx).
Disk #7: Linking Loader & RLOAD, Harkness
Disk #8: CRTSET, Lanpher (May '82)
Disk #9: DATECOPY, DISKFIX9 (Aug '82)

NOTE: All are as published or received by 68 Micro Journal, some have fixes and patches.

This is a reader service only! No Warranty is offered or implied, they are as received and are for reader convenience ONLY. Also 6800 and 6809 programs are mixed, as each is fairly simple (mostly) to convert to the other.

PRICE: 8" Disk \$19.95 - 5" Disk \$17.95

68 MICRO JOURNAL
 POB 794
 Hixson, TN 37343
 615-842-4600

* indicates 6800, ** indicates BASIC SWTPC or TSC - 6809 no indicator.

MASTER CARD - VISA accepted - Foreign add sufficient postage surface or air!!

WINCHESTER FOR MOTOROLA EXORCISOR/MDOS

☐ 10 MB Winchester hard disk runs MDOS on Motorola Exorcisor System ☐ No modification to MDOS required
☐ MDOS based software stays alive ☐ All user software operates without modification ☐ Optional SA-801R flexible diskette drive system ☐ Optional 10 MB removable cartridge.

CSA For information call (714) 566-3911
Computer System Associates
 7562 Trade Street, San Diego, CA 92121



'68' Micro Journal

STYLOGRAPH

6809
 WORD PROCESSING SYSTEM

STYLOGRAPH 2.0

All of the convenience and features for which Stylograph is well known plus:

- True proportional printing on specialty printers.
- Files longer than memory.
- "Help" command to aid in learning.
- New menu driven, self prompting functions.
- Left and right scrolling for pages larger than screen.
- Embedded printer control commands allowed.
- Simplified method for underline, bold-face, superscript, etc.
- Supports NEC, Diablo, Qume, 737, and 739 printers.
- Can be user configured for virtually any terminal or printer.

\$295, manual \$15, updates from old versions \$180.

STYLOGRAPH MAIL MERGE

This program takes files of variables, such as names and addresses, and inserts them into a Stylograph text file for automated mail list generation. It will also allow a number of Stylograph text files to be appended at printout time so that page numbers and headings will be continuous in the printout.

\$125, manual \$10.

STYLOGRAPH SPELLING CHECKER

This is a valuable addition to any word processing application. It checks all words in a manuscript against an internal dictionary. The dictionary included has a vocabulary of over 36,000 words and is fully expandable. New words encountered in the text may be added to the dictionary making the creation of custom tailored dictionaries a snap.

\$145, manual \$10.

When ordering specify operating system (FLEX™, Uniflex™ or OS-9) and disk size. VISA & MC accepted.

20% discount on 3 program order.

OS-9 Version Now Available



STYLO SYSTEMS

P.O. BOX 916 IDAHO FALLS, IDAHO 83402
 208-529-3210

OmegaSoft Puts Pascal Power In Program Development

Finally. A fast, efficient way to design, code, debug and maintain your microprocessor software. OmegaSoft's Pascal. The software development tool that takes the sting out of programming. For industrial or business use, large or small scale.

How does it work? Simple. We've compiled Pascal into the compact native code that is most common to professional applications. OmegaSoft Pascal is designed to integrate easily with existing software development tools available on most 6809 operating systems.

Single Pass Punch

Our compact single-pass compiler quickly translates Pascal into 6809 assembly language code. This means less time-wasting disk swapping and fewer multiple passes, with greater success rates. OmegaSoft Pascal is based on the proposed ISO standard with compatible extensions designed to make Pascal an important tool in industry and business. Byte wide variables allow convenient access to I/O devices. Custom I/O devices can be used in place of normal Pascal INPUT and OUTPUT files.

Practical Yet Precise

OmegaSoft's Pascal long, 32-bit integer support also allows representation of money amounts without speed penalties of BCD implementations. And our 7 digit real format is compatible with AM9511 APU with options to use this chip instead of software routines.

OmegaSoft Pascal's random access file is excellent for fast access of data for business applications programs and other mass storage uses. Sophisticated applications programs are simplified with dynamic length strings, such as:

- Process control.
- Communication systems.
- Robotics.
- Automatic test systems.
- Microcode generation.
- Scientific computation.
- Development system software.

High Speed, High Performance

OmegaSoft Pascal generates high speed code without sacrificing ROMability, position independence or reentrancy. Modular compilation is supported along with several methods of interfacing to assembly language variables and routines, including interrupt procedures.

OmegaSoft Pascal's runtime library provides that only the modules necessary to execute the Pascal program are loaded, which translates into much smaller object modules and reduced memory costs.

Our symbolic debugger interfaces with the assembly language output of the compiler for fast, accurate program execution. Debugger commands allow other exciting possibilities:

- Displaying and changing variables using Pascal names.
- Tracing through statements.
- Setting breakpoints at the start of Pascal statements.

Features

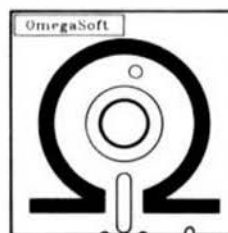
- Compatible with MDOS™ and XDOS™, FLEX™, DOS69™, and OS-9™.
- Entry, External, and Absolute attributes for variables and procedures.
- Optimized in-line code generated wherever possible rather than subroutine calls.
- Compiler generates 6809 assembly language – can either be used with the debugger or assembled and linked to form a loadable object module.

- Standard data types plus long-integer, hex, dynamic length string, and device.
- Expanded operations for pointers and characters allows low level code to be written in Pascal.
- Extensive type conversion functions.
- Compiler – Requires 48K system.
- Symbolic Debugger to run on any of the operating systems.
- Utilities to assist in creating a "CHAIN" file for automatically linking the user program with the runtime library.
- Manuals – detailed language reference handbook and a configuration manual.
- Runtime library and source.
- ISO validation report available.
- Debugger allows display and change of variables using their symbolic names and stepping through the program one statement at a time.

Single unit prices start at \$425 (domestic).

Support products include: OmegaSoft's Motorola compatible Relocatable Assembler/Linker and for higher speed OmegaSoft's Arithmetic Processor option uses the AMD9511 to support integer, longinteger, and real calculations for Pascal runtime. Available soon will be a multitasking kernel and a high speed translator between compiler output and relocatable object code.

TM: MDOS, XDOS, Motorola trademarks; Flex, TSC trademark; DOS69, Smoke Signal Broadcasting trademark; OS-9 Microware trademark.



OmegaSoft

P.O. Box 70265
Sunnyvale, CA 94086
Telephone (408) 733-6979

FLEX* and UNIFLEX* Software

SUPER SLEUTH DISASSEMBLER \$99-FLEX \$100-UNIFLEX

This program processes 6800/1/2/3/5/6/9 6809 programs, enabling the user to analyze, modify, and disassemble (with tables) object code, with output to terminal, printer, and disk, and cross-reference and label-definition capabilities.

Z-80/8080/5 SUPER SLEUTH DISASSEMBLER \$99-FLEX \$100-UNIFLEX

This version of SUPER SLEUTH processes Z-80/8080/5 object code on the 6800/1/9.

CROSS-ASSEMBLERS each \$50 3/\$100-FLEX each \$60 5/\$120-UNIFLEX

These programs and TSC macros enable the user to process 6800/1, 6805, 6502 Z-80/8080/5 programs in original format.

6805 and 6502 DEBUGGING SIMULATORS each \$75-FLEX \$80-UNIFLEX

These programs enable the user to interactively analyze, modify, and debug 114 6805 and 6502 object code.

6502-TO-6809 XLATOR SYSTEM \$75-FLEX \$80-UNIFLEX

This program enables the user to translate 6502 assembly code into 6809 assembly code, noting inexact conversion.

6800-6809 and 6809 PIC/PID XLATORS both \$50-FLEX \$60-UNIFLEX

These programs enable the user to translate 6800/1 assembler programs to 6809 machine code and to convert 6809 programs to position-independent code and data, using PC, S, U, X, and Y as base registers.

UNIFLEX SIMULATOR FOR FLEX \$100-FLEX \$110-UNIFLEX

This program enables the user to debug UNIFLEX assembler programs using the TSC DEBUG and other facilities of FLEX.

FULL SCREEN FORMS DISPLAY (6809 X-BASIC) \$50-FLEX \$75-UNIFLEX

These programs enable the user to define and generate table-driven full-screen display and data-entry program.

FULL SCREEN MAILING LIST (6809 X-BASIC) \$100-FLEX \$110-UNIFLEX

These programs enable the user to define and maintain mailing-list-oriented data bases.

FULL SCREEN INVENTORY/MRP (6809 X-BASIC) \$100-FLEX \$150-UNIFLEX

These programs enable the user to define and maintain inventories, and include hierarchical materials requirement planning.

TABULA RASA SPREADSHEET (6809 X-BASIC) \$100-FLEX \$200-UNIFLEX

These programs enable the user to generate and maintain tabular compilation schemes, providing a simple user interface and sophisticated report-generation, similar to DESKTOP PLAN (TM Desktop Computing).

TSC BASIC/XPC UTILITY PROGRAMS all \$25-FLEX \$50-UNIFLEX

These programs enable the user to resequence or cross-reference any Basic program and generate XPC Basic sort programs.

Computer Systems Consultants, Inc., 1454 Latta Lane, Conyers, Ga 30207, Telephone Number 404-483-1717/4570

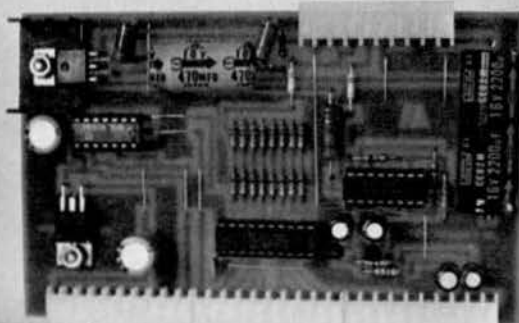
Programs in source on disk - specify 5 1/4", sides, density.
Detailed printed manuals provided with all products.
For VISA and MASTER CARD give account, exp date, phone
U S funds only - add 6% (10% foreign) for shipping
Open Purchase Orders for D and B rated clients only.
Exclusive U K dealer Computerware in London
Call or write for catalog and dealer information.
* Technical Systems Consultants

MUSIC BOARD

8 bit DAC with LOW-PASS FILTER and AUDIO AMP for SS-30 BUS

Disk contains 1 hour of music and 2 programs to enter songs

Assembler source code included



8" or 5" FlexB 375.00 Check or Money Order

Please add 5% sales tax

PALM BEACH SOFTWARE

3648 LANTANA ROAD 305 987-3660

LANTANA FL 33462

If your needs include any of the following...

MAILING LISTS	INVENTORY	INQUIRY
ACCOUNTING	CHECKBOOKS	CASH FLOW
SCHEDULING	ORDER STATUS	MARKET TRENDS
SERVICE CONTRACTS	SAMPLE & ANALYSIS	SPREAD SHEETS
RECORD SELECTION	REPORT GENERATION	POSTING
ON-LINE UPDATING	BATCH PROCESSING	TRACKING

...You really should consider DMS2/VM

The DMS2/VM Data Management System provides maximum flexibility, yet it is simple to operate. You are immediately in command. Start by defining a simple application, such as a mailing list and graduate quickly to sophisticated analysis or batch processing applications. Define databases up to 24 fields. Update interactively or input from formatted text files. Generate output reports or databases to your specifications. Merge information. Select, sort and reformat records. Do line calculations. Total columns. Subtotal groups of related lines. Format output into a predefined form. Create output data files.

Virtually all standard data management functions are at your command, through an English-like control language, which may be entered interactively or from a control file. The syntax even allows for variable parameters, for which you are prompted at execution time.

DMS2/VM offers many transparent features such as data compression, automatic reorganization and database backup. The Virtual Memory technology used allows a host of operations on files as large as 4 Megabyte which are cumbersome or impossible with other methodologies. The user is spared these technical concerns, and thus is free to concentrate on the application. Implementation time is reduced to minutes, as opposed to days or weeks for comparable applications written in any programming language.

DMS2/VM can save you time and money. Over 3000 R&D hours went into making DMS2/VM the most flexible and versatile DMS in the business. Put it to work for you! It is priced to be affordable at only \$99.95.

Written in modular assembler, requires 32K and two drives, FLEX O/S, Add P&H \$2-50. Foreign \$7.50. N.Y. State add sales tax. Specify 5 or 8".

also available: Basic Accounting System adaptable to a variety of environments. DMS peripherals and user interface. Word processing and general purpose utilities. Call or write for latest information.

WESTCHESTER Applied Business Systems
Post Office Box 187
Briarcliff Manor, N.Y. 10510

Technical Consultation: 914-941-3552 (6-18 P.M. EST - or leave message)
Sales Representative: 212-697-8488 (Ask for Sandy)

FLEX is a trademark of Technical Systems Consultants, Inc.



INFOMAG DBMS

A data base management system specifically designed for microprocessor based computer systems.

- You specify new databases and screen entry masks.
- Define and work with subgroups of the master database.
- Sort groups.
- Audit trail can be maintained.
- Password protection.
- Control code definition for different terminals and printers.
- Columnar reports, modular reports, reports drawing from primary and secondary files.
- Statistics reports.
- Database files can be accessed by userwritten BASIC programs.

FLEX - \$295.00 UniFLEX - \$395.00

OSBORNE BUSINESS PROGRAMS

This ENHANCED IMPLEMENTATION of the Osborne and Associates Business Programs is the only implementation available with the full capability of the original Wang Minicomputer version.

FEATURES INCLUDE:

- * KEYED FILES to eliminate slow searches and sorts.
- * PASSWORD and MASTER PASSWORD PROTECTION to limit unauthorized access to your business data.
- * LINKED can be run linked to G/L.

ACCOUNTS RECEIVABLE This package is an open invoice system, it prints aging reports, monthly statements, open item listings, etc. You can maintain trade discount schedules and scheduled penalties for late payment. FLEX.....\$295 UniFLEX.....\$395

ACCOUNTS PAYABLE This package is an invoice-linked system, it prints aging reports and will print out your checks. FLEX.....\$295 UniFLEX.....\$395

GENERAL LEDGER Accepts postings to the various accounts from external sources: Accounts Payable, Accounts Receivable, and Cash Journal. The normal posting is double entry to reduce off-balance posting errors. FLEX.....\$295 UniFLEX.....\$395

6809 SOFTWARE TOOLS

6800 RRMAC (Relocatable Recursive Macro Assemblers)

These are true Macro assemblers with external linkage capability. Variables may be local, global, or external. One assembler produces relocatable 6800 code which is loaded with the included Linking Loader. The second assembler, a cross assembler runs on your 6800 and produces relocatable 6809 object code from existing 6800 or new 6809 source files. Two assemblers plus a Linking Loader are including in this package.

FLEX.....\$150

6809 RRMAC (Relocatable Recursive Macro Assemblers)

One assembler produces relocatable 6809 code which is loaded with the included link editor. The second 6809 assembler, a cross assembler, runs on your 6809 and produces relocatable 6800 code.

FLEX.....\$150

SUPER SLEUTH DISASSEMBLER, enables the user to examine and/or modify object program files on disk or in memory on 6800/1/9 systems.

FLEX.....\$99

GREAT PLAINS COMPUTER CO.

P.O. BOX 916 / IDAHO FALLS, ID 83402 / PH: (208) 529-3210
Visa and MasterCard accepted.

HOME CONTROL

HCOM software provides TOTAL control of ALL 236 possible devices. We provide the software and an adapter cable, you provide the RADIO SHACK PLUS N-POWER(tm) controller (026-1182) and a parallel port. This real time software may be adapted for whatever CPU speed you may be running and allows parameters to be modified for noisy electrical environments. Remember that NO ultrasonic links are required. Two NEW commands have been added to the standard six commands allowing you to LOCK and UNLOCK all devices under your control.
\$SAVAILABLE FOR PERCOM AND STANDARD DB9 FORMAT NOW!!
REQUIRES APPROXIMATELY 1K TO RUN

5" DISK WITH ADAPTER & DOCUMENTATION-----\$35.00

OPB DUAL PORT SERIAL BOARD

COMPATIBLE WITH BOTH 8530 & 8530-C
FULL RTS, DCD, CTS etc. JUMPER CONTROL
ALL CHIPS SOCKETED--USES 1488/1489 DRIVERS
ALL SUPPLY VOLTAGES REGULATED (+5, +12, -12)
TWO DB25 CONNECTORS ON BOARD FOR EASY HOOKUP

COMPLETE & TESTED WITH DOCUMENTATION-----\$49.00

TG-1 BIT RATE GENERATOR

9 STANDARD BIT RATES FROM 110-19.2K SIMULTANEOUSLY
USES NO PARTS FROM YOUR PRESENT SYSTEM
FACILITATES UPGRADING TO 8830C STATUS

COMPLETE & TESTED WITH DOCUMENTATION-----\$49.00

GOLD PLATED CONNECTORS ARE STANDARD
PLUS N-POWER is a trademark of RADIO SHACK.
CTS is a trademark of MICROWARE SYSTEMS CORP.
TG112 is a trademark of GIMEX INC.

TEXAS COMPUTER

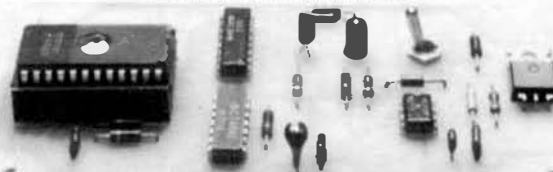
add for shipping \$3.00

P.O. BOX 120818 ARLINGTON, TX 78012

VISA / MC 817-275-1848

EPROM PROGRAMMER

Shown assembled. EPROM not included.



For single supply 2516, 2716 & 2758 EPROMs. Connects through a user supplied interface to any computer system. Interfacing requires two 8-bit ports plus hand-shake lines. One of the ports must be software controllable for input or output. Timing is done via hardware, thus is independent of MPU clock rate. Verify erased. Program — entire or partial. Auto verify after programming. Transfer contents to RAM for modifying or duplicating.

Select Documentation for:

6502
6800
6809
8080/8085/Z80

Interface to:

6820 PIA or 6522 VIA
6820 PIA
6820 PIA
8255 PPI

Comprehensive documentation booklet contains schematic, instructions for construction, check-out and use, and a well commented assembly listing for the specified MPU.

Complete kit of parts (includes ZIF socket) \$ 45.00

Bare PC board and Documentation \$ 25.00

Software listings for additional MPUs
(with purchase of kit or PC board) \$ 5.00

Ordering: Specify MPU. Add 5% for P&H. Overseas add 10%. Ariz. residents add 5% tax.

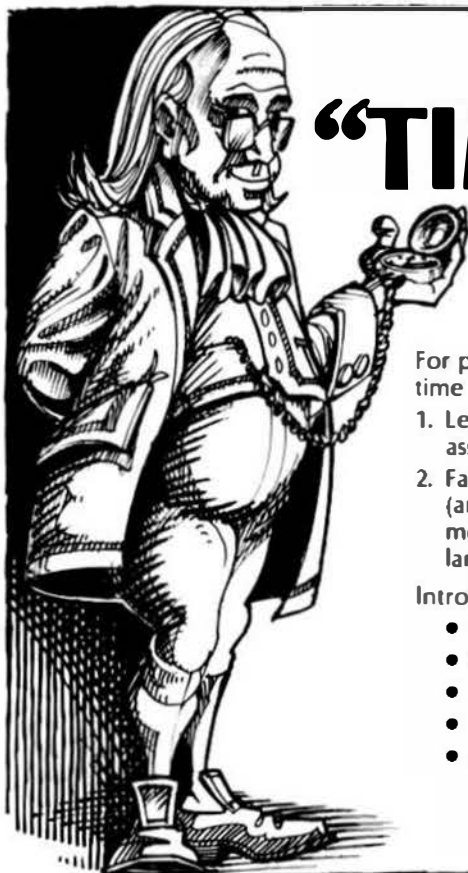


Micro Technical Products, Inc.

814 W. Keating Ave., Dept. J

Mesa, Arizona 85202 • 602-839-8902





"TIME IS MONEY"

INTROL-C for the 6809

For process control, Introl-C saves time in two important ways:

1. Less development time than assembly language
2. Faster execution time (and smaller code size) than most other high-level languages

Introl-C/6809 includes:

- C Compiler
- 6809 Assembler
- Linking Loader
- Library Manager
- Standard Library

Introl-C produces efficient, re-entrant, position-independent, ROMable object code.

Available as resident software for: FLEX-09* \$300, UNIFLEX* \$350, and OS-9** \$300 or as cross-software for CP/M*** \$350 systems. 8" SSD diskette only. Add 10% outside U.S. and Canada.

Trademarks:

*Technical Systems Consultants

**Microware, Inc.

***Digital Research

INTROL CORPORATION
647 W. Virginia St.
Milwaukee, WI 53204
(414) 276-2937

*OS9 Application Software

Specialty Electronics, Inc.

GENERAL LEDGER with CASH JOURNAL

The general ledger is the center of the Specialty Electronics Interactive Accounting System. With the package you can:

1. Produce balance sheets and income statements in various formats
2. Define account names, spacing, positioning, headings, and subaccounts
3. Format special reports and print percentages
4. Post by hand, cash journal or by using the interactive accounts receivable payable and payroll
5. Provide a clear audit trail for all entries
6. Input data in an easy to follow format
7. Use for multi company accounting without modification

General Ledger I-code \$399.00

ACCOUNTS RECEIVABLE

Your Accounts Receivable can be followed with a minimum of time investment using these features:

1. Regular invoicing, debit and credit memos, full and partial payments
2. Progressive billing and payments
3. Aging of periods specified by the user
4. New customers entered as needed
5. Statements are generated listing individual invoices and overdue amounts totaled by aging category
6. Total interaction with the general ledger with lgs, shipping, and travel expenses computed separately and posted to various accounts

Accounts Receivable \$799.00

INVENTORY

The Specialty Electronics Interactive Accounting System Inventory Control Package provides the tools for complete control of a large and active inventory, including:

1. Reports for quantities on hand, quantities on order, activity and many other categories
2. Complete item description, category groups, supplier information, order dates, reorder quantities, etc.
3. Simple input and reconciliation procedures

Inventory Control I-code \$299.00

PAYROLL

with

COST ACCOUNTING COMING SOON

ACCOUNTS PAYABLE COMING SOON

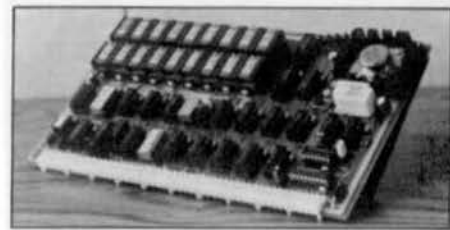
Complete Documentation \$ 19.95

*OS9 and Basic09 are trademarks of Microware, Inc. and Motorola Corp.

(405) 233-1632 • P.O. Box 541 • 2110 W. Willow • Enid, OK. 73701

QMM1

Quarter Megabyte Memory



The **QMM1** is a "Quarter Megabyte" (256K) dynamic parity memory board for SS50-C 6809 systems with 20 bit addressing.

- Full 2Mhz operation with transparent on board refresh, runs continuously at 2Mhz with no cycle stretching, stealing or halting.
- Optional parity feature halts processor and sounds audible alarm upon detecting a read error.
- Switch disable of up to 82K, 4K, or 8K blocks in any or all banks.
- Gold bus connectors, socketed. Professional quality board.
- On board transient surge and ESD protection.
- All boards assembled, burned in, tested and warranted for 1 year.

NOTES: Not for 6800 systems. Requires 1 jumper addition to 6809 CPU board.

	config. #	without parity	config. #	with parity
64K	110	\$ 495.00	221	\$ 550.00
128K	130	\$ 695.00	241	\$ 775.00
192K	350	\$ 935.00	461	\$1045.00
256K	37	\$1135.00	481	\$1270.00

ORDER FROM:

D.P. Johnson
7855 SW Cedar-
crest St.
Portland, OR
97223

(503) 244-8152

Delivery:
2-4 weeks ARO

TERMS: Cash, Check, or Krugers with order, add \$5.00 shipping and ins. in USA, foreign orders include shipping for 1.5 kilos.

PRODUCTS FOR YOUR RADIO SHACK

COLOR COMPUTER

NEW!

MACRO-80C

The Micro Works is pleased to announce the release of its disk-based editor, macro assembler and monitor, written for Color Computer by Andy Phelps. THIS IS IT — The ultimate programming tool!

The powerful 2-pass macro assembler features conditional assembly, local labels, include files and cross referenced symbol tables. MACRO-80C supports the complete Motorola 6809 instruction set in standard source format. There are no changes, constraints or shortcuts in the source language definition. Incorporating all of the features of our Rompack-based assembler (SDS80C), MACRO-80C contains many more useful instructions and pseudo-ops which aid the programmer and add power and flexibility.

The screen-oriented text editor is designed for efficient and easy editing of assembly language programs. The "Help Key" feature makes it simple and fun to learn to use the editor. As the editor requires no line numbers, you can use the arrow keys to position the cursor anywhere in the file. MACRO-80C allows global changes and moving/copying blocks of text. You can edit lines of assembly source which are longer than 32 characters.

OCBUG is a machine language monitor which allows examining and altering of memory, setting break points, etc.

The editor, assembler and monitor — as well as sample programs — come on one Radio Shack compatible disk. Extensive documentation included. MACRO-80C Price: \$99.95

SDS80C — Our famous editor, assembler and monitor in Rompack. Complete manual included. Price: \$89.95

PARALLEL PRINTER INTERFACE — Serial to parallel converter allows use of all standard parallel printers. You supply printer cable. P180C Price: \$69.95

MICROTEXT — Get printouts while using your modem! Also download to cassette. General purpose terminal Rompack. Price: \$59.95

THE MICRO WORKS

Also available: Machine language Monitor ★ 2-pass Disassembler ★ Books ★ Memory Upgrade Kits ★ Parts and Services ★ Call or write for information

P.O. BOX 1110 DEL MAR, CA 92014

California residents add 6% Tax
Master Charge/Visa and
COD Accepted

714-942-2400

YOU NEED COLOR FORTH!!

Why?

- Forth is faster to program in than Basic
- Forth is easier to learn than Assembly Language
- Forth executes in less time than Basic

Forth is a highly interactive language like Basic, with structure like Pascal and execution speed close to that of Assembly Language. The Micro Works Color Forth is a Rompack containing everything you need to run Forth on your Color Computer. Color Forth consists of the standard FORTH Interest Group (FIG) implementation of the language plus most of FORTH-79. It has a super screen editor with split screen display. Mass storage is on cassette. Color Forth also contains a decompiler and other aids for learning the inner workings of this fascinating language. It will run on 4K, 16K, and 32K computers. Color Forth contains 10K of ROM, leaving your RAM for your programs! There are simple words to effectively use the Hi-Res Color Computer graphics, joysticks, and sound. The 112-page manual includes a glossary of the system-specific words, a full standard FIG glossary and complete source listing. **COLOR FORTH... THE BEST!** From the leader in Forth, Talbot Microsystems. Price: \$109.95

GAMES

Star Blaster — Blast your way through an asteroid field in this action-packed Hi-Res graphics game. Available in ROMPACK, requires 16K. Price: \$39.95

Pac Attack — Try your hand at this challenging game by Color Computerware, with fantastic graphics, sound and action! Cassette requires 16K. Price: \$24.95

Berserk — Have fun zapping robots with this Hi-Res game by Mark Data Products. Cassette requires 16K. Price: \$24.95

Adventure — *Black Sanctum* and *Calixto Island* by Mark Data Products. Each cassette requires 16K. Price: \$19.95 each.

Cave Hunter — Experience vivid colors, bizarre sounds and eerie creatures in hot pursuit as you wind your way through a cave maze in search of gold treasures. This exciting Hi-Res game by Mark Data Products requires 16K for cassette version. Price: \$24.95

Dickson Computer Systems

Distributors in

ONTARIO CANADA

For Gimix and SWTPC

Dealers for:

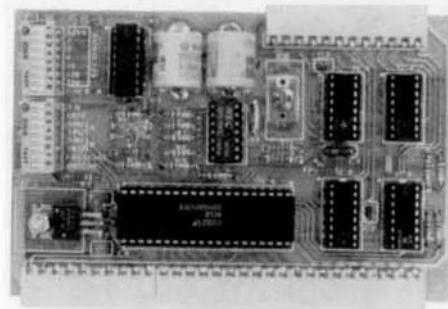
Qume and Centronics

Custom designed systems and
packaged software available now



Dickson Computer Systems
2600 John Street, Unit 116
Markham, Ontario L3R 3W3
Canada (416) 475-2311

CALENDAR-CLOCK / TIMER / PARALLEL PORT



Calendar - Clock

CLK68-1

- Keeps date and time whether or not the computer is on
- All clock functions software controlled
- On card battery (included) and charging circuit runs for months
- Day of week, month/day/year, hour:minute (12/24 hr)

Interval Timer

- Set preset 999.999, auto-loading, etc.
- Countdown also 0.1 to 9.99 sec
- 555 mode features 100-1000 Hz frequency sweep or SWTPC 484
- Countdown interrupt interval time 1 sec to 99.99 sec

Parallel I/O Port

- Fully buffered 8 bit parallel data
- DIP switches select input or output buffering (on computer and the board)
- Compatible with parallel printer drivers or built-in version of BASIC

Construction

- Fully assembled, solder plated, 6 pins screened

Manual

- Well documented - 36 pages

Order & PCB layout available

Assembled and tested	\$119.95	Kit	\$89.95
Goldplated bus conn	7.50	2 MHz option	2.50
Disk 5 or 8 in. SSB or Flex*	OS-9 Available NOW		14.95

* OS-9 is a trademark of Microsoft Systems Corporation
* Also a trademark of technical Systems Consultants, Inc.

ROBERTSON ELECTRONICS
1003 Warm Sands Dr. SE
Albuquerque, NM 87123

Phone (505) 294-0025
NM residents add 4% tax
Add \$3 Shipping & Handling

6800/6809 FIRMWARE
HUMBUG - The Ultimate Monitor. Multiple breakpoints, single-stepping, formatted memory dumps, I/O control. For CPU boards made by Elektra, Glens, Mellin, Percom, Star-Kits and SHTP, with or without video boards, \$40 to \$75.

6800/6809 SOFTWARE
SPELL 'N FIX - Finds mistakes and fixes them too, using its dictionary of almost 20,000 words. For Flex or Percom DOS, \$89.29. (Order SSB versions from Alford and Associates).

WRITE 'N SPELL - access a 20,000-word dictionary right from your text editor and become an expert speller. For TSC's Editor and Flex \$75.17, other versions coming soon.

CHECK 'N TAX combines checkbook reconciliation with income tax breakdowns in a way you'll appreciate every April 15th. Available for Flex or Percom DOS, \$50.

BASIC UTILITY PACKAGE renames, pretty-prints, cross-indexes and more. For Percom DOS or MiniFlex, \$40.

SORT-MERGE - the only one for Percom disk systems, \$30.

NEUTALK makes your computer talk to you. This memory dump program is ideal for checking memory contents against a printer listing. \$30 on disk or cassette.

ELIZA - Our machine language version is just super. For Flex or Percom DOS or cassette, \$15.

THREE-DEE is three-dimensional tic-tac-toe, for Flex or Percom DOS or cassette, \$15.

6800/6809 HARDWARE
SBC-02 single-board computer uses 6802 with RAM, ROM, I/O, ideal controller, intelligent interface and more. PC board \$25, controller kit \$75, kit with HUMBUG \$115, kit with Basic \$135.

CT-PS serial/parallel interface card for RS-232 terminal and/or parallel keyboard. Ideal for video board systems. PC board \$25.

COLOR COMPUTER SOFTWARE
HUMBUG is great! Enter and debug programs, analyze tapes, connect to remote terminals or computers, do things nobody else can with HUMBUG. \$39.95 on disk or cassette, \$69.95 on ROM pack.

SPELL 'N FIX (see above) for Color Computer disk systems \$89.29.

CHECK 'N TAX (see above) for Color Computer disk systems \$50.

NEUTALK (see above) on disk or cassette, \$20.

SHRINK (Eliza) or **OXRO (Othello)** on disk or cassette, \$15 each.

RENOTERM - Connect a CRT terminal to the Color Computer and run it remotely, even through a modem. \$19.95 on disk or cassette.

LPRINT - use a non-standard printer with your Basic. Disk or cassette \$19.95.

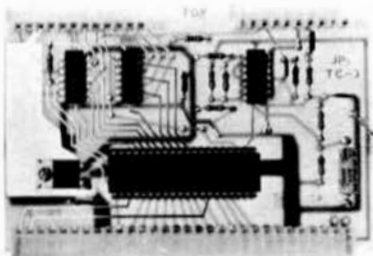
For further information, send or call for catalog, or order by phone. NY State residents please include sales tax.

STAR-KITS

P. O. BOX 209
 MT. KISCO, N.Y. 10549
 (914) 241-0287

JPC PRODUCTS FOR

6800 COMPUTERS



High Performance Cassette Interface

- **FAST** - 4800 Baud Loads 4K in 8 Seconds!
- **RELIABLE** - Error Rate Less Than 1 in 10⁶ Bytes
- **CONVENIENT** - Plugs Directly into The **SWTPC**.
- **PLUS** - A Fully Buffered 8 Bit Output Port Provided
- **LOW COST** - \$59.95 For Complete Kit.
- **OPTIONAL** - CFM/3 File Manager

Manual & Listing \$19.95
 (For Cassette Add) \$ 6.95

TERMS CASH, MC or VISA

Shipping & Handling \$3.00



JPC PRODUCTS CO.

Phone (505) 294-4623

12021 Paisano Ct.

Albuquerque, N.M. 87112

'68' MICRO JOURNAL

★ The only ALL 6800 Computer Magazine.

★ More 6800 material than all the others combined:

MAGAZINE COMPARISON

(2 years)

Monthly Averages

KB	BYTE	6800 Articles CC	DOBB'S	TOTAL PAGES
7.8	6.4	2.7	2.2	19.1 ea. mo.

Average cost for all four each month: **\$6.53**

(Based on advertised 1-year subscription price)

'68' cost per month: **\$2.04**

That's Right! Much. Much More

for About

1/3 the Cost!

OK, PLEASE ENTER MY SUBSCRIPTION

Bill My: Master Charge ☐ — VISA ☐

Card # _____ Exp. Date _____

For ☐ 1-Year ☐ 2 Years ☐ 3 Years

Enclosed: \$ _____

Name _____

Street _____

City _____ State _____ Zip _____

My Computer Is: _____

68 Micro Journal
 5800 Cassandra Smith Rd.
 Hixson, TN 37343

SUBSCRIPTION RATES

USA

1 Year \$24.50, 2 Year \$42.50, 3 Year \$64.50

*FOREIGN SURFACE Add \$12.00 per Year to USA Price

*FOREIGN AIRMAIL Add \$36.00 per Year to USA Price

**CANADA & MEXICO Add \$5.50 per Year to USA Price
 Cash (USA) or drawn on a USA Bank!!!



UNIVERSAL DATA RESEARCH INC.

INTRODUCING SOFTWARE FOR THE COLOR COMPUTER AND TRS 80 MODEL III

Data Base Manager - \$150
Church Contributions Pkg. - \$150
Single Entry Ledger - \$95

STAND ALONE SOFTWARE:

- * SINGLE ENTRY LEDGER
- * CHECK BALANCER
- * COSTING & ESTIMATING
- * CHURCH CONTRIBUTION PKG.
- * CHURCH SCHOOL BILLING.
- * FUND RAISING
- * CUSTOM PROGRAMMING
- * M.F.I. (Extensive Mfg./P.O.)
- * BULK MAILINGS
- * BULK MAILINGS
- * MAINTENANCE SCHEDULE
- * VENDOR PROGRAM
- * PURCHASE ORDER
- * ACCOUNTS PAYABLE
- * CUSTOMER PROGRAM
- * ORDER ENTRY
- * ACCOUNTS RECEIVABLE

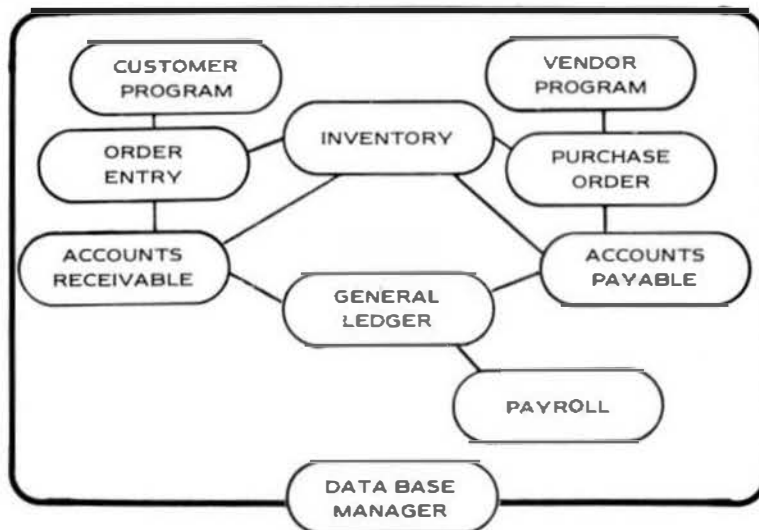
VISA

MasterCard

FLEX and UniFLEX SOFTWARE FOR THE 68XX OPERATING SYSTEM

INTEGRATED BUSINESS PROGRAMS
for Large and Small Companies

ACCOUNTS RECEIVABLE	\$395	\$495
ACCOUNTS PAYABLE	\$395	\$495
GENERALS LEDGER	\$395	\$495
INVENTORY 2	\$395	\$495
PAYROLL	\$395	\$495
DATA BASE MANAGER	\$450	\$550



2457 WEHRLE DRIVE
BUFFALO, NEW YORK 14221
(716) 631-3011

MASTER CHARGE AND VISA ACCEPTED

FLEX and UniFLEX are Trademarks
of Technical Systems Consultants

64K SS-50 STATIC RAM

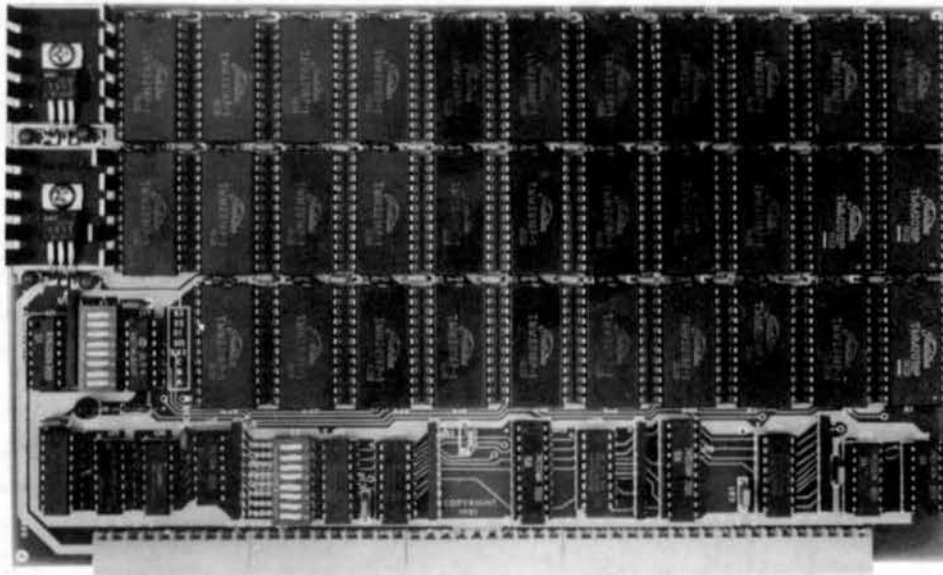
\$259⁰⁰
(48K KIT)

NEW!

NEW!

**LOW
POWER!**

**RAM
OR
EPROM!**



**BLANK PC BOARD
WITH DOCUMENTATION
\$52**

**SUPPORT ICs + CAPS - \$18.00
FULL SOCKET SET - \$15.00**

ASSEMBLED AND TESTED ADD \$40

FEATURES:

- ★ Uses new 2K x 8 (TMM 2016 or HM 6116) RAMs.
- ★ Fully supports Extended Addressing.
- ★ 64K draws only approximately 500 MA.
- ★ 200 NS RAMs are standard. (TOSHIBA makes TMM 2016s as fast as 100 NS. FOR YOUR HIGH SPEED APPLICATIONS.)
- ★ Board is configured as 3-16K blocks and 8-2K blocks (within any 64K block) for maximum flexibility.
- ★ 2716 EPROMs may be installed anywhere on Board.
- ★ Top 16K may be disabled in 2K blocks to avoid any I/O conflicts.
- ★ One Board supports both RAM and EPROM.
- ★ RAM supports 2MHZ operation at no extra charge!
- ★ Board may be partially populated in 16K increments.

56K Kit	\$309
64K Kit	\$359

16K STATIC RAMS?

The new 2K x 8, 24 PIN. static RAMs are the next generation of high density, high speed, low power, RAMs. Pioneered by such companies as HITACHI and TOSHIBA, and soon to be second sourced by most major U.S. manufacturers, these ultra low power parts, feature 2716 compatible pin out. Thus fully interchangeable ROM/RAM boards are at last a reality, and you get BLINDING speed and LOW power thrown in for virtually nothing.

Digital Research Computers

(OF TEXAS)

P.O. BOX 401565 • GARLAND, TEXAS 75040 • (214) 271-3538

TERMS: Add \$2.00 postage. We pay balance. Order under \$15 add 75¢ handling. No C.O.D. We accept Visa and MasterCard. Tex. Res. add 5% Tax. Foreign orders (except Canada) add 20% P & H. Orders over \$50, add 85¢ for insurance.

ARCADE 50

POWERFUL COLOR GRAPHICS

Uses the new TMS9918A Video Display Processor. High resolution 256 x 192 pixel display with 16 colors. 16k Bytes of onboard RAM does not reduce user memory. 32 graphic images can be individually moved with simple X-Y commands for smooth animation.

External video input allows subtitling.

NTSC composite video output

SOUND EFFECTS AND MUSIC

Three AY3-8910 Programmable Sound Generators

Nine simultaneous voices

Three independent noise sources

Onboard stereo amplifier drives two 8 ohm speakers

ADDITIONAL I/O CAPABILITIES

Eight analog inputs with 8 bit resolution

Supports four joysticks with pushbutton switches

Eight bit parallel I/O port

Entire unit maps into 256 bytes of memory

DOCUMENTATION AND SOFTWARE

Programming manuals for Video and Sound Processors

Subroutine library and Super Demo Maze Game

Example programs in BASIC, FBASIC and ASSEMBLY

User library and sales support

ARCADE 50, assembled and tested	\$325.00
Video and Audio connector set	15.00
4 Joystick connector set	15.00
2 Radio Shack Joysticks	24.00
UHF channel 33 modulator	32.00
Gold Molex connectors	12.00
A/BASIC for 6800	110.00
FBASIC for 6800	110.00
FBASIC (with ARCADE 50)	75.00
ARCADE 60 (TRS MODEL I)	395.00
LABVIDEO (Motorola EXORbus)	375.00
NEW MYDS 6809 Processor Board	225.00

* Comes assembled with PIA and ACIA

* 12 Sockets for 2716, 2732 or RAM

* Supports DMA disk I/O

* Ideal for 6809 upgrade or process control

TERMINUS DESIGN INC
16 SCARBROUGH ROAD
ELLENWOOD, GA 30049
(404) 474-4866

PRODUCTS IN THE MIX:
RGB ARCADE 50 ADAPTOR
ARCADE 100 (S-100)
DMA Disk Controller

FBASIC

TERMINUS DESIGN INC., in conjunction with Microware Systems Corporation, is proud to announce FBASIC - an enhancement of Microware's 6800 A/BASIC. Their fast compiled BASIC has been adapted for 6809 users with added video and sound features for ARCADE 50 users. FBASIC is a true compiler that produces optimized machine language modules which are ROMable and require no Run-Time package. FBASIC requires less memory overhead and runs hundreds of times faster than BASIC interpreters. It supports standard BASIC instruction including String functions, Disk I/O and fast integer arithmetic with multiple-precision capability. Graphics verbs and functions fully support the ARCADE 50. Arcade statements include:

INIT	MODE	BLANK	BACKDROP
SIZE	MAG	VREG	DELAY
MOVE	DRAW	FCOLOR	JSWITCH
REMOVE	RDRAW	BCOLOR	SWITCH
PSG	-tone	ENVL	VOLUME
ADC	SPRITE	SPNAME	ENDEF
SPCOLOR	RSPRITE	SPDEF	PATDEF
VPEEK	VPOKE	VPRINT	

Specify 5" or 8" soft sector disk for TSC's FLEX or MICROWARE'S OS/9 system.
TERMS: CASH, VISA, MC, C.O.D.

5 MEG HARD DISK \$600⁰⁰

This is a new Shugart SA-10002 5.33 megabyte hard disk drive. Interfacing is a snap with the Western Digital WD1000-80 intelligent controller card (\$500). An example of the interface and software can be found in this month's issue. Power supply (\$150) is good for 2 drives and controller. Interfaces for CP/M, S-100, and STD BUS are also available.



105 S. Main St. Greer, SC 29651
Ph. 803-877-7471

ENGINEERS/TECHNICIANS THE MICRO 68000 IS DESIGNED FOR YOU!

COMPLETE, READY-TO-GO SYSTEM INCLUDES:

- ☐ 6 amp switching power supply
- ☐ Keyboard
- ☐ Display - Hex & Binary
- ☐ Pete Bug keyboard monitor
- ☐ Optional Macs Bug CRT monitor
- ☐ Attractive cabinet
- ☐ Dual RS232 interface
- ☐ 32 bit parallel I/O
- ☐ Versabus compatibility
- ☐ The only system that provides for direct entry of 68000 machine code.



For information call (714) 566-3911
Computer System Associates
7562 Trade Street, San Diego, CA 92121



ELEKTRA CABINET

Made of heavyweight 0.090" thick aluminum.
Interior Size: 18-1/2" wide by 21-7/8" deep x 6-3/4" high
Heavy duty A.C. line cord.
A.C. fuse holder
EMI filter
Fan with filter
Back panel has 10 cutouts for "D" type data connectors.
Front panel has key on/off power switch, 2 illuminated push buttons (Reset and NMI/Abort), and 2 cutouts for 5-1/4" floppy drives.

Without power supply:	\$260.00
With 110v power supply, disk regulator board or two filter plates:	\$460.00
With 220v power supply, disk regulator board or two filter plates:	\$510.00

POWER SUPPLY

Highest quality linear power supply conservatively rated at 15a @ 8v.
3a @ 16v 3a @ -16v
3 primary inputs to adjust for light, medium, and heavy loading
110v version: \$175.00 220v version: \$225.00

DISK REGULATOR BOARD with cables

Powers 2 5-1/4" floppy drives: \$50
Filter (fan): \$10.00

ELEKTRA CPU 8/9

Choice of 6802, 6808 or 6809 CPU
(6802 and 6808 are software compatible with the 6800 or the opcode level).

DEVICE	6809 ADDRESS	6802-6808 ADDRESS
3 2716 Eprams	Epram #0 F800-F7FF	F800-F7FF and EC00-E7FF
	Epram #2 F000-F7FF	F000-F7FF
	Epram #1 E800-E7FF	E800-E7FF
1K Static RAM	E400-E7FF	A400-A7FF and A000-A3FF
MC6840 Triple Timer	E210-E217	8200-8207

MC14411 Baud Rate Generator producing baud rates of:
Low Range 110, 130, 300, 600, 1200, 4800, and 9600
High Range 480, 600, 1200, 2400, 4800, 19200, and 38400

The board does not contain a DAT and does not support extended addressing.
The board supports DMA by either HALT or BUSREQ when a 6809 CPU is used.

DMA to the devices on the CPU card is not supported.
The board will run any of the MIKBUG™ compatible monitors in the 6802-6808 mode and 30UG-E, HUMBUG, and GAMBUG-09 in the 6809 mode. The ELEKTRA CPU 8/9 will run any of the popular disk controller boards with the appropriate software. Special versions of OS-9™ L1 are available.

Base board: \$50.00* Kit: \$225.00* Assembled: \$275.00

ELEKTRA DPS Dual Port Serial Card

Fits the standard 30 pin SS-50 bus I/O slot.
Can be configured for 4 addresses per port with the B port 2 addresses higher than the A port or for 16 addresses per port with the B port 4 addresses higher than the A port.

Each port is terminated at two 16 pin dip sockets, one socket configured for modem and the other socket configured for terminal or printer. RTS, CTS, DTR, DCD, DSR are appropriately implemented.

Each port has independent selection of baud rate.
Each port allows the interrupt request to be jumpered to the IRQ or FRQ/NMI bus line.

Base board: \$20.00* Kit: \$60.00* Assembled: \$80.00
Assembled cable (two required for each interface board): \$20.00 each

ELEKTRA DPP Dual Port Parallel Card

Fits the standard 30 pin SS-50 bus I/O slot.
Can be configured for 4 addresses per port or 16 addresses per port (occupying the first four addresses of the I/O slot).

The direction of the TTL buffers can be controlled by either on board jumper connection or by a signal from the peripherals.

The interrupt request lines for each port may be individually jumpered to the IRQ or FRQ/NMI bus line.

Base board: \$20.00* Kit: \$60.00* Assembled: \$80.00
Assembled cable (two required for each interface board): \$20.00 each

AAA Chicago Computer Center

120 Chestnut Lane • Wheeling, IL 60090 • (312) 459-0450

Phone consultation available most weekdays from 4 PM to 6 PM

ELEKTRA COMPUTER PRODUCTS



ELEKTRA Motherboard

Heavyweight 0.125" thick 18" long by 9" wide.
11 memory (30 pin) slots. 4 or 8 slots may be cut off for shortening to 14" and 10" lengths respectively.
8 I/O (30) pin slot.

1" spacing between all memory and I/O slots.
On board baud rate generator with high and low ranges providing jumper selectable rates for each of the five I/O baud rate lines.

Complete address decoding and selectability for the I/O ports.
Choice of 4 K or 16 addresses per I/O port.

Slow device circuitry permitting 4 MHz 30 pin disk controller to run with 2 MHz 50 pin CPU boards.

Extended addressing capability for meeting SS-50C bus specifications.

Base board: \$80.00 Kit: \$240.00** Assembled: \$300.00**

**Gold square pin connectors instead of tin add \$80.00

ELEKTRA Chassis

Includes cabinet, power supply, disk regulator board and power cables, and assembled motherboard with gold connectors, totally installed. Ready to use with documentation. \$895 Add \$50.00 for 220V.

GEMIX 16K memory board with control registers \$195.00
GEMIX #15 memory board with 32K static RAM \$295.00

*WARNING

AAA Chicago Computer Center does not provide repair or diagnostic service for customer assembled kits. AAA Chicago Computer Center does warranty and maintain service for our assembled boards. The customer should carefully take into consideration the small differential separating our kit and assembled prices when making his choice of purchase.

We have introduced our line of computer equipments with the purpose of offering the highest quality of components possible at affordable prices. These products are intended for OEM applications where it is the responsibility of the purchaser to integrate these components with suitable memory, disk controller(s), drives, and software along with I/O terminal(s) to form working computer system(s).

SMOOTH™ Software

SUPER MODEM PROGRAM

Transmit manually to distant computer

Transmit disk files (text) of any length to distant computer

Receive and save disk files (text) of any length on local disk system. If sending computer does not support an X-on/X-off protocol, then the received files are limited in size by the computer memory.

Teletype to transmit and receive text at speeds up to 9600 baud. (CRT terminal) must be capable of operating at a baud rate higher than the one the modem is operated at.) Half duplex option in case distant computer doesn't echo.

Echo option so user can simulate a time sharing system. (Super Modem Program doesn't support auto-answer but the source is provided for those individuals who wish to adapt our program to their special needs.)

Replaces CR with CR/LF (user option) for those using time sharing systems that don't transmit LF's.

Slow disk file transmit (user option) based on character verification for use on time sharing systems to which disk files cannot be sent at speed suggested by the baud rate.

Please specify 6800 SSB, 6800 FLEX™, or 6809 FLEX™, 5" or 8" Manual and disk with both source and object code \$75.00

STANDARD MODEM PROGRAM

Same as Super Modem Program above but without ECHO option, CR/LF for CR option, slow disk file transmit option, nor X-on/X-off option. Reception of disk files is limited to those small enough to completely fit within the receiving buffer.

Please specify 6800 SSB, 6800 FLEX™, or 6809 FLEX™, 5" or 8" Manual with instructions, source listing, and flow chart; disk with both source and object code \$45.00

ALL IN ONE

Editor - Text Processor - Mailing Labels

Mailing Lists - Use any CRT terminal and printer

Supports Editing commands such as bottom change, delete, find, insert (single line), input (multiple lines), list, next, overlay (with cursor editing, character deletion and insertion), overwrite (for selected data or text), print, restart, set, top, underline up and verify.

Supports Text Processing commands such as block copy, block move, centering, margin justification (width and narrow), paging, and tabbing.

Mailing Lists and Labels. Use the same mailing list disk file (with protected areas) for both mailing labels and repeat letters. Repeat letters are personally addressed to each person or selected persons on the mailing list.

Most Powerful File Manager found in any editor. Append one file to the end of another, or insert (merge) one file into another as designated by the line pointer. Print specified buffer to your printer or to a disk file. Edit files larger than the text buffer. Does not produce output files when not desired. Delete disk files from the editor.

Printer commands. Control characters can be sent to the printer for format control either directly from the control terminal or by imbedding them in the text. The set command contains interface initialization and character output routines to support the SWTPC MP-C interface as well as the standard serial and parallel interfaces. Jumps are also provided to user supplied printer routines. User selects the port address 10 thru 7, A or B thereby eliminating the need for the user to install printer software routines. Editor can be initialized for either 4 or 18 addresses per port.

Editor allows access to either the monitor or DOS and then reenter (Warm Start) without destroying previously prepared text in the buffer. The Restart command erases contents in the buffer without the user having to reload the Editor.

The Editor allows the user to toggle between full duplex (no echo) and half duplex (echo) as needed. It responds to commands in both upper and lower case and can be used to create assembler source code and Basic programs as well as text.

Specify 6800 SSB, 6800 FLEX™, 6809 FLEX™, 5" or 8" Printed source listing is available for an additional 35.00

Software by Technical Systems Consultants, Inc.

FLEX™ (Includes Editor and Assembler) 150.00

UniFLEX™ (Includes one year maintenance and update) 450.00

Editor 50.00

Assembler 50.00

68000 Cross Assembler on 6809 250.00

6809 Cross Assembler on 6800 100.00

Text Processor 75.00

Extended Basic 100.00

Basic Precompiler (Specify standard or extended) 50.00

Pascal (FLEX™) 200.00

Pascal (UniFLEX™) (Add \$75.00 for one year's maintenance and update) 225.00

Sort/Merge Package 75.00

6800 Files™ Utilities 75.00

6800 Files™ Utilities 100.00

Debug Package 75.00

Diagnostic Package 75.00

Software by Microware Systems Corp.

OS-8™ Level One Operating System 75.00

OS-8™ Level Two Operating System 75.00

OS-8™ Macro Text Editor 300.00

OS-8™ Interactive Assembler 300.00

OS-8™ Interactive Debugger (Disk version) 100.00

CIS Cobol Compiler 250.00

Pascal Compiler 100.00

SWTPC Kit Assembled

DMS2 Disk Controller Board (NEW) 595.00

DMS2 Disk Controller Board (Used, very limited quantity) 450.00

SWTPC FLEX™ Disk and manual 35.00

SWTPC FLEX™ Disk without manual 10.00

DC-2 Disk Controller (Limited Quantity) 125.00

4K RAM Board (Limited Quantity) 40.00

SBUG-E (2716 compatible) 19.95

SWTBUG (6830 compatible, limited quantity) N/A

MP-A2 6800 CPU Board 150.00

MP-S2 Serial interface (dual port) 120.00

MP-LA Parallel interface (dual port, limited quantity) 40.00

MP-L2 Parallel interface (dual port) 120.00

MP-R Single voltage 2716 prom programmer 114.50

MP-N Calculator board 54.95

MP-T Interrupt timer 92.00

MP-BM 8K 4054 Memory board (limited quantity of kits) 150.00

S32 Universal Static Memory Board 124.50

MP-09 6809 CPU board 295.00

6809 Chassis P.S. 6809 CPU 8K RAM One Serial Port 799.00

Universal BASIC Bare Motherboard, 6800/6809, 4/18 addresses per port, 8.50 pin/8 30 pin slots, baud rate generator, 15 1/8" x 9 3/8" 85.00

..... 80.00

ELEKTRA Motherboard (Bare)

Connectors (10 pin, Titanium-Tin plated 5 microns for near gold quality)

Mate with square cross section pins each 50

Female each 75

Gold, Mate requires pins or female each 150

SMOOTH™ and ELEKTRA™ are trademarks of AAA Chicago Computer Center

FLEX™ and UniFLEX™ are trademarks of Technical Systems Consultants, Inc.

OS-8 and BASICOS are trademarks of Microware Systems Corp.

GIMEX™ and GHOST™ are registered trademarks of GIMIX Inc.

ELEKTRA™ SSSO Computer Products

ELEKTRA Dual drive cabinet for 5-1/4" drives with power supply, line cord, fuse, power switch, and power cable to drives 150.00

ELEKTRA Dual drive cabinet, power supply, ps cable for 8" drives 350.00

Cabinet for dual 8" drives only 250.00

Power supply for dual 8" drives only 120.00

PS cables only (Specify brand and type of 8" drives) 30.00

Special Software

4K 6809 HUMBUG 75.00

4K 6800 HUMBUG (RAM needed at \$A000 and \$D000) 65.00

2K 6800 HUMBUG (With cassette LOAD and PUNCH) 40.00

2K 6800 HUMBUG (Extra commands instead of cassette software) 40.00

Other HUMBUG versions including video versions are available

Spell'n Fix by Peter Stark 89.29

Dynamite Disassembler 60.00

SUPER STEUTH Disassembler System 98.00

DISK DRIVES 30 day guarantee, single and double density capability

5-1/4" single head, 40 tracks \$250.00

5-1/4" single head, 40 tracks, floppy 260.00

5-1/4" double head, 40 x 2 tracks 335.00

5-1/4" single head, 80 tracks 335.00

5-1/4" single head, 80 tracks, floppy 345.00

5-1/4" double head, 80 x 2 tracks 470.00

MPI - Service Manual 20.00

Siemens Manual 10.00

8" single head, 77 tracks 375.00

8" single head, 77 tracks, Thin-Line 450.00

8" double head, 77 x 2 tracks 495.00

8" double head, 77 x 2 tracks, Thin-Line 525.00

Microtime 6800 Calendar and Clock Board (assembled and tested) 105.00

Barcode, connector, and documentation only of above 35.00

(See review Feb 1980 68 Micro Journal)

Microtime II 75.00

Data Mail 16K EPROM bareboard 12708 chip(s) 30.00

Printers

Epson MX-80 (Centronics compatible parallel interface) 439.00

(with Serial RS-232 interface option) add 75.00

Spare Print Head 39.95

Spare ribbon cartridge 15.00

C. H. Comp 125 cps, 9 x 7, bi-directional serial or parallel 295.00

Quintad Microline 82A 120 cps, 9 x 9, bi-directional serial and parallel 495.00

Tractor for 82A 70.00

Optimal Technology, Inc. EP-2A-79 Eprom Programmer 169.00

(Personal Computer extra for above programmer)

Optimal Technology, Inc. 30 pin parallel I/O board for EP-2A-79 35.00

Software package for EP-2A-79 (Specify 6800 or 6809) 29.95

EP-2A-79-M-01 Eprom Programmer (User configurable and supplies power) 79.95

Single Signal Broadcasting

Monitor on 2716 with manual (Specify Chetam or SWTPC) 75.00

DCB-1A Double Density Controller Board for 5" and 8" with DOS 549.00

DOS580 DOS Update and documentation only of above 75.00

LMB-1A Motherboard 399.00

SCB-09 6809 CPU Board 399.00

M-16-X 16K Static Memory Board 195.00

M-24-X 24K Static Memory Board 295.00

M-32-X 32K Static Memory Board 395.00

GIMIX

2 MHz 6809 Plus CPU, time of day clock, battery backup, 1K NMOS RAM 578.00

CMOS RAM substitution 8.00

GIMIX Dynamic Address Translator 35.00

SWTPC compatible DAY 15.00

Making cycle detect card 39.23

Disk Controllers (All have data separators and can be used with either single or double headed drives)

5" single density controller without 1771 chip 158.38

5" single density controller complete 198.48

5" and 8" single density controller complete 226.58

5" double density controller with variable precomp 298.28

DMA 5" AND 8" double density controller with variable precomp 588.68

GIMIX version of FLEX™ (without Editor and Assembler) 90.00

Double disk regulator card 66.22

Ribbon cable for two 5 1/4" disk drives (short) 34.96

Ribbon cable for two 5 1/4" disk drives (long) 44.96

Ribbon cable for two 8" disk drives (long) 44.26

8" disk drive cabinet with power supply 848.18

Memory CMO WITH NMOS NO

BAT BACKUP BAT BACKUP

16K Static RAM Board with control registers 195.00

32K Static RAM Board with 32K of RAM installed 295.00

*discontinued, limited quantity available

64K Static RAM Board with 24K of RAM installed N/A

64K Static RAM Board with 32K of RAM installed 348.27

64K Static RAM Board with 48K of RAM installed 398.37

64K Static RAM Board with 56K of RAM installed 518.47

64K Static RAM Board with 64K of RAM installed 578.57

16 Socket EPROM/ROM/RAM Board 638.67

8K Prom board (2708) 98.34

4K PPD 4K Prom Board and 2708 Prom Burner 198.00

I/O Boards

Single port 30 pin serial interface (Requires 1 cable set) 88.41

Dual port 30 pin serial interface (Requires 2 cable sets) 128.43

8 port 30 pin serial interface with baud rate generator 318.46

Dual port 30 pin parallel interface (Requires 2 cable sets) 88.42

8 port 30 pin parallel interface with interrupt generator 198.45

Cable sets for above boards, specify board 24.95

2MHz 6809 PLUS Computer System with 56K Memory 2498.29

Above System with #58 Controller and Special Software Pkg 2988.59

Above System with #58 Controller and Special Software Pkg 3248.49

*with CMOS RAM and Battery Backup add 300.00

Mainframe (Cnassis, PS, Switches, Fan, Motherboard, Baud Rate Gen) 1198.19

Shipping and handling estimates

Within the Continental U.S. please add 3% (\$5.00 minimum)

Foreign, prepaid and add 10% (\$10.00 minimum) for light items only

Heavy items must be prepaid and will be shipped Emvay Air Freight Collect

Please phone during consultation hours if questions arise regarding shipping fees

Master Charge, Visa and American Express honored

AAA Chicago Computer Center

120 Chestnut Lane, Wheeling, IL 60090

(312) 459-0450 Phone Consultation available most weekdays from 4 PM to 6 PM

Our apology: We are running 4 weeks behind answering our mail regarding technical questions. Please phone during consultation hours for technical help

Prices and inventory are subject to change without advance notice

This ad is our catalog

GRANITE COMPUTER SYSTEMS

FLEX 9 DISC AVAILABILITY

Granite Computer software now available on 5.25 FLEX discs

THE DISASSEMBLER FAMILY

Source listings identical with TSC 6809 EDITOR - User symbol tables - Local and global labels and expressions - Optional generation of occurrence numbered local (program) labels - Easy identification of data areas - FCB - FDB - FCC - Step disassembly one program or data statement at a time - Source tape or disc for TSC EDITOR input - Run TSC ASSEMBLER with minimal modification - Problem codes flagged on output

Convenient menu driven options carry out tedious error prone disassembly operations - rapidly and accurately

JUST WHAT YOU NEED TO CONVERT THOSE 6800 & 6502 PROGRAMS!

6800 to 6809 DISASSEMBLER (see July '68' ad) \$49.95
6502 to 6809 DISASSEMBLER (see August '68' ad) \$49.95

COMPANION PROGRAM

6809 to 6809 DISASSEMBLER (see June '68' ad) \$49.95

LIMITED OFFER

Any two DISASSEMBLERS ordered together \$74.95
All three DISASSEMBLERS ordered together \$99.95

.. Others in the series of super programs for the 6809 ..

EPROMMER - use with SWTPC MP-R Programmer \$39.95

TEXTWRITER - use with TSC EDITOR - synergistic editing and processing package \$39.95

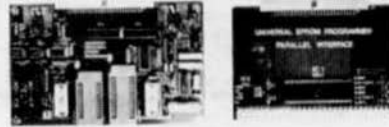
FILEMANAGER - use with JPC TC-3 high speed I/O board - comprehensive cassette oriented operating system - Cassette/Disc \$29.95 2716-1 EPROM \$39.95

All efficient - well documented and VERY FRIENDLY

Run on any 5550 6809 system with minimal change - Comprehensive Manuals - Object programs on KC cassette or 5.25 FLEX discs

M/C GRANITE COMPUTER SYSTEMS
Route 2 Box 445
Hillsboro, NM 83244 VIBA

ALL - IN - TWO



* Probably the most versatile EPROM PROGRAMMER available. Interfaces & software for EXORCISOR-II (fully addressable) and 15-30 bus systems.

* PROGRAMMER AND VISI/IES 2508/2708, 2516/2716 (SINGLE AND TRI-VOLT TYPES) 2532, 2732, 2732A, 2564, 2764 AND THE 1284 TRS2528 (16K & 8K) -> -> -> WITHOUT ADDITIONAL 'PERSONALITY' MODULES <- <- <-

* PROGRAMMER extends out to your work area via 5' of twisted pair cable.

* EXTENSIVE COMMANDS MENU.....MOVE BLOCKS OF DATA, READ, PROGRAM, AND VERIFY PROM'S, EXAMINE/CHANGE BUFFER, FORMATED DUMP OF BUFFER, FILL BUFFER WITH SPECIFIED BYTE, RETURN TO DOS OR MONITOR.

* Fully documented user's manual w/schematics & theory of operation.

* Software source file included on disc.....enables customizing.

* Professionally finished PCB's w/solder resist & component overlay.

5-30 INTERFACE/PROGRAMMER/BASEPLATE (bare boards).....\$85.00
EXORCISOR INTERFACE/PROGRAMMER/BASEPLATE (bare boards).....\$95.00
HARD TO GET PARTS KIT (SPECIAL DEL SW, 12M, 1M RES, trimmer).....\$17.00

5-30 INTERFACE/PROGRAMMER/BASEPLATE/CABLE (assembled).....\$295.00
EXORCISOR INTERFACE/PROGRAMMER/BASEPLATE/CABLE (assembled).....\$375.00

-> -> -> PRICES INCLUDE SHIPPING VIA REGISTERED AIR MAIL <- <- <-

PAYMENT BY: INT'L MONEY ORDER, BANK DRAFT, VISA OR MASTER CHARGE.
(PERSONAL CHECKS TAKE 4 -> 6 WEEKS TO CLEAR!)

SPECIFY: CPU (6800/6809), DISK SIZE (512K), AND DOS (SSB/FLEX/ROS/DS-9)

MEMPHIS MICRO SYSTEMS LIMITED
NORTHEAST LABORATORIES
NORTH WALSHAM, WIMBORNE
ENGLAND, NR26 9SA

TEL: (0492) 405109
TEL: 07360 SHURET 8

WE ARE A STOCKING DISTRIBUTOR OF SSB, BIRIS, TSC AND MICROWARE.



MIDDLE-C

WW Small-C09 Release 2.0

All features of level 1.0, plus logical operators, assignment operators, conditional compilation, and an improved user interface. (Version 2.1 planned for OS-9.) Includes source to linkage editor (RLOAD 3.1) on separate disk. Run-time library source included. 48K recommended. Requires version 2 of TSC's assembler for FLEX9.

INTRODUCTORY PRICE:

For FLEX9 (includes RLOAD 3.1) \$80.00
(If you already have RLOAD) \$72.50
For DOS69D Inquire

WW Small-C 2.0 PRICE GOES UP ON NOVEMBER FIRST!
SAVE \$20 BY ORDERING NOW

Purchase of level 2.0 includes FREE update to 2.1!

RLOAD 3.1 separately \$17.50
Update from RLOAD 3.0 SASE

OTHER ITEMS OF INTEREST:

Chuck's Corner, by Dr. Charles Adams, published sporadically.

Eight issues (nominally one year) \$ 2.00
With 8 #10 SASE's \$ 7.50
Without SASE's

The C Programming Language

by Kernighan and Ritchie

Software Tools, and

Software Tools in Pascal

by Kernighan and Plauger

Books with any software order \$14.00 ea
ordered separately \$17.00

FUTURE PLANS:

A relocating macro assembler (to be included with release 2.2 of WW Small-C09), a screen-oriented editor (written in C), LISP (maybe), other applications in a public-domain C user's library, and a continuation of our unusually liberal update policy.

We are looking for software authors. Please inquire about our requirements and royalty schedule, before submitting software.

The Fine Print

Unless otherwise specified, all software is supplied on FLEX-format, 35-track, single-sided 5" disk. Prices good until November 1st, 1982. Shipping via first class mail is already included, except add \$2.00 for orders shipped outside North America. Add \$35 for "overnight" parcel service to Canada, which still takes at least three days, or \$12 for "express mail" in US. Add \$2 handling for Visa/MC. Allow 3 weeks for non-certified check. Purchase order must be accompanied by payment. Texas residents: add \$0.25/disk. Release 2.1 of WW Small-C09 is scheduled for 1st quarter '83. The phone number is for our answering service. You may call to request further information to be mailed, or place a Visa/MC order 8:30-4:30 weekdays (CDT). For fastest response to technical questions, send #10 SASE.

FLEX is a trademark of TSC. OS-9 is a trademark of Microware.

word's worth

P.O. Box 28954
Dallas, Texas 76228
(214) 321-9285

DB-25 RS-232

PORT PIC U1 74sc244 **

PORT PIOA U2 74sc244 **

PORT PIOD U3 74sc244 **

PORT BDC U4 74sc373 **

PORT POD U5 74sc373 **

PORT POE U6 74sc373 **

PORT POP U7 74sc373 **

XTAL 3.579 U8 MC146803E2P

U9 74sc138 U10 74nc00 U11 74nc00 U12 74sc139

20519 U13 74nc00 U14 74nc00 U15 CD4078

U16 741s244 ** U17 741s244 ** U18 741s640 **

U19 EPROM / RAM 2716 / 6116

U20 EPROM / RAM 2716 / 6116

U21 EPROM / RAM 2716 / 6116

U22 EPROM / RAM 2716 / 6116

U23 74sc373

55 - 50 BUSS CONNECTOR

1982 THOMAS INSTRUMENTATION CPU-5

SOFTWARE

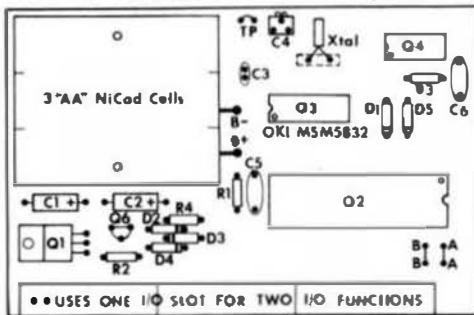
CHIPS

HARDWARE

Extender Cards assemb. with logic aid	
SS-50/50C	\$ 35.00
SS-30	\$ 25.00
SP-1 Prototype Board A/T	\$195.00
SP-1 Bare Card	\$ 49.00
SS-50 Wire Wrap Board B/C	\$ 39.00
SS-30 Wire Wrap Board B/C	\$ 20.00
6802 Super CPU A/T	\$235.00
6802 Super CPU B/C	\$ 59.00
Video RAM, B/C \$49.00, A/T	\$195.00
Parallel I/O, B/C \$49.00, A/T	\$139.00
SS-30 Backplanes 4,6,8,12,16 position	
@ \$5.00 per slot w/o connectors	
SS-30 8 pos. BP w/o connect.	\$ 39.00
Transition Card \$49.00, A/T	\$ 95.00
Molex Gold Male 1,40, Female	\$ 1.60
Molex Tin Male 1,40, Female	\$.30

59

Model 6800CL4 CalClock/TIMER



IT'S A HARDWARE CALENDAR/CLOCK

- Keeps date and time without servicing by the computer
- Day-of-week, month/day/year, hour:min:sec (12/24hr. + auto Leap Year)
- Hands off setting/control/access of ALL functions via software
- On-card battery and charging circuit keeps time for months, power off

WITH AN INTERVAL TIMER INCLUDED

- For (TSC/Flex2 compatible) printer spooling, multi-tasking, etc.

Fully assembled & tested	\$ 99.95	\$*Disk(Flex2 □ Flex9 □)*	\$ 10.00
Complete kit*	\$ 69.95	Goldplated buss connectors	\$ 6.00
Bare board*	\$ 35.00	Shipping & handling	\$ 3.00

* FULLY DOCUMENTED: instructions; diagrams; theory; more than 20 pages of sample software (automatically puts date in Flex2/9 date buffer, adds time-of-day to assembly listings, maintains constant, current time + date display on top line of CRT). Batteries not included. All IC's socketed.

© FLEX is the registered trademark of Technical Systems Consultants, Inc.



COMPUWARE Corporation
P.O. Box 2710
Cherry Hill, NJ 08003
09-428-2309

New Jersey buyers: ADD 5%
Terms: CASH; MC; or Visa
Flex9 □ Flex2 □ (default) □
UniFLEX □

F&D Associates
1210 Todd Road
New Plymouth, Ohio
45654

614 - 592 - 5721

Send for free Catalog

Visa ~ Master Charge ~ C.O.D.

AGA-1 VIDEO PORT

Add video capabilities to your S50 system inexpensively. Based on SYNERGEX SY6545. 4k screen memory (byte-wide). 2k or 4k EPROM resident character generator. Occupies one slot of 3C pin bus. Uses only 4 addresses, making it compatible with older as well as newer systems. No "snow" or other glitches. 80 x 24, 64 x 16, and other formats possible. 5 x 7, 6 x 8, 7 x 9, and other character styles possible. Pseudo-graphics by pixel patterns in character generator EPROM. Software driver routines provided in the form of assembled source listings and also available on floppy diskette.

ASK ABOUT MAJOR PARTS KIT and EPROM SYSTEM MONITORS

AGA-1 bare board and documentation \$39.00

5 x 7 character generator. Upper and lower case. No descenders. As few as 8 raster lines per character row. Makes it easy to get 24 lines on low cost monitors.

CH5X7E 2716 EPROM character generator \$20.00

Diskette containing 6800 source and binary of drivers and binary of CH5X7E at 0600 ready to program into EPROM.

AGADSK1 specify 5 1/4 or 8 inch diskette \$20.00

Add \$3 per order s/h. Ohio res. add 5 percent

SOFTWARE FOR THE HARDWARE

** TOOLS FOR PROBLEM SOLVERS **

- oo FIRST -- You have a problem -- OH WOEE!
- oo SECOND -- O! course! Use a computer!
- oo THIRD -- Choose the best hardware -- a 6809!
- oo FORTH -- Choose the most useful software.

----> FORTH - A TOOL FOR CRAFTSMEN!

- > Join the thousands of problem solvers who have discovered the FORTH method of producing results, instead of impediments.

†FORTH is a refined version of FORTH Interest Group standard FORTH for 6809 (and 6800); 50% faster than FIG-FORTH, several times faster than BASIC.

FORTH is unique among computer languages in many respects, not the least of which is that it was created by problem solvers to help them on with their tasks, rather than by computer scientists.

FORTH applications have spanned a wide range of tasks -- listening to galaxies, talking with dolphins, running robots, controlling production line machinery, and sophisticated graphics systems.

Users of FORTH report productivity gains of 2 to 10 over other development tools. †firmFORTH(tm) is for the programmer who needs to squeeze the most into RAM.

†FORTH and firmFORTH are trademarks of Talbot Microsystems.

©FLEX is a trademark of Technical Systems Consultants, Inc.

TALBOT MICROSYSTEMS 1927 Curtis Ave., Redondo Beach, CA 90278 (213) 376 9941

†FORTH™ THE PROFESSIONAL'S CHOICE from the author of 6809 fig-FORTH TALBOT MICROSYSTEMS

----> †FORTH SYSTEMS <---

For all FLEX systems: GIMIX, SWTP, SSB, or EXORCISOR; or convert to other systems. Specify 5 or 8 inch diskette, hardware type, and 6800 or 6809. For standalone versions, write.

Manuals available separately - price in (). Add \$5/system for shipping, \$12 for foreign air.

** †FORTH - extended fig FORTH (1 disk) \$100 (\$15) with fig line editor.

** †FORTH+ - extended more! (3 5" or 2 8" disks) \$250 (\$25)

Includes 2nd screen editor, assembler, extended data types utility vocabularies, GOING FORTH CAl course on FORTH, gams, and debugging aids.

** TRS-80 COLORFORTH - available from The Micro Works

----> APPLICATIONS PROGRAMS <---

** firmFORTH - 6809 only. \$550 (\$10)
For target compilations to ramable code. Automatically deletes unused code and unneeded dictionary information. Includes full source code for target compiler and essential FORTH nucleus. Requires but does not include †FORTH+.

** TINY PASCAL compiler in FORTH. 6800/09 \$75 (\$20)

** FORTH PROGRAMMING AIDS - elaborate decompiling and program analysis tools \$150 (\$10).

** Also available: code for floating point, timers, and real time programming.

Compilers and **C**ross compilers

TELECON'S C COMPILERS OFFER YOU

- FULL C
- UNIX* Ver. 7 COMPATABILITY
- NO ROYALTIES ON GENERATED CODE
- GENERATED CODE IS REENTRANT
- C AND ASSEMBLY SOURCE MAY BE INTERMIXED
- UPGRADES & SUPPORT FOR 1 YEAR

IN THESE CONFIGURATIONS:

HOST	6809 TARGET	PDP-11*/LSI-11* TARGET	8080/(Z80) TARGET	8088/8086 TARGET
FLEX*/UNIFLEX* 6809	\$200.00 WITHOUT FLOAT \$350.00 WITH FLOAT	500.00	500.00	500.00
RT-11*/RSX-11* PDP-11*	500.00	200.00 WITHOUT FLOAT 350.00 WITH FLOAT	500.00	500.00
CP/M* 8080/(Z80)	500.00	500.00	200.00 WITHOUT FLOAT 350.00 WITH FLOAT	500.00
PCDOS*/MSDOS* 8088/8086	500.00	500.00	500.00	200.00 WITHOUT FLOAT 350.00 WITH FLOAT

Others Pending

C SOURCE AVAILABLE FOR \$2,500⁰⁰

SO . . . IF YOU'RE READY TO MOVE UP TO C ...

CALL

408-275-1659

TELECON SYSTEMS

90 E. Gish Road, Suite 25
San Jose, CA 95112

*PCDOS is a trademark of IBM CORP. MSDOS is a trademark of MICROSOFT. UNIX is a trademark of BELL LABS. RT-11/RSX-11/PDP-11 is a trademark of Digital Equipment Corporation. FLEX/UNIFLEX is a trademark of Technical Systems consultants. CP/M is a trademark of Digital Research.

IBMPACK for UniFLEX

Adapt your UniFLEX system to mainframe computer systems!

FORMAT/READ/WRITE IBM 3740 DISKETTES

- formatbm formats diskettes to IBM standard format.
- readbm displays directory of an IBM diskette copies files from IBM diskettes to UniFLEX files.
- writelbm copies UniFLEX files to IBM diskettes.
- dumpbm displays the hexadecimal value of a sector on an IBM diskette.

ASCII-EBCDIC conversion can be overruled.

Very simple to apply.

ANDRO-DATA

Kantonstrasse 100

CH 6048 HORW

Switzerland



\$ 125.—

Check or VISA
Europe add \$ 5.00 S&H
Overseas \$ 20.00 S&H

Please specify your UniFLEX serial-number on your purchase order.

UniFLEX is a trademark of Technical Systems Consultants.

Special Introductory Offer

MAC*EDIT

and

MAC 95 for only \$80.00

MAC*EDIT — 6809 Full Screen Editor

- "What you see is what you get" Cursor Oriented Editing
- Written for the non-programmer.
- Designed for the Electric Window and is easily configured for other memory mapped video cards.
- 33 editing functions
- Includes the SHUFFLE utility for moving data between files.
- Uses option files for saving and restoring left/right margins and tab settings.
- Print function does titling and paging; single, double and triple spacing; prints up to 99 copies.
- Allows concurrent printing and editing of files without an interrupt timer.
- Reference Manual Included
- Requires 20K memory and FLEX 9 \$50.00

MAC 95 — 6809 MACRO PREPROCESSOR

- Assembler programming made easy.
- Allows programmers to create new instructions tailored to their applications.
- Supports position — independent and re-entrant programming techniques.
- Allows recursive macro calls.
- Has 5 addressing modes available: direct, indirect, literal, immediate and register.
- Includes fully commented source.
- Provides 39 macros including general macros, file management macros, and memory table macros.
- Reference Manual and source listings included.
- Requires 16K memory and FLEX 9. \$50.00

MAC*EDIT and MAC 95 supplied on 5 1/4" FLEX 9 disk

MAC SOFTWARE

404/476-5113

P.O. Box 1129, Duluth, GA 30136

Check, M.O. accepted. Add \$2.50 postage and handling. GA residents add 3% sales tax.

FLEX is a trademark of Technical Systems Consultants, Inc. Electric Window is a trademark of Pe com Data Company, Inc.

XDATA

DUB

UNIFLEX* DATA TRANSMITTER

XDATA ALLOWS UNIFLEX BASED SYSTEMS TO TRANSMIT/RECEIVE FILES TO/FROM OTHER COMPUTER SYSTEMS VIA MODEM. IE. CP/M, MAIN FRAMES, OTHER UNIFLEX SYSTEMS.

- VERIFIES TRANSMISSION INTEGRITY USING CHECK-SUM OR CYCLIC REDUNDANCY CHECK
- AUTOMATICALLY RETRANSMITS BAD BLOCKS
- TRANSMITS DATA IN 128 BLOCK BYTES

UNIFLEX* BASIC DECOMPIER

DUB CREATES SOURCE STATEMENTS FROM PROGRAMS COMPILED WITH TSC BASIC AND RUNS UNDER TSC UNIFLEX.

- SAVE \$*: DO YOUR OWN PROGRAM CHANGES
- EFFICIENT AND EASY TO USE
- DECOMPILES ALL VERSIONS OF TSC BASIC
- AUTOMATICALLY DETERMINES WHICH BASIC IT IS DECOMPIING
- ROUTE OUTPUT TO TERMINAL OR DISK

BUSINESS DATA SYSTEMS

27 GARY COURT
DAYTON, N.J. 08810

\$179

BOTH
PROGRAMS

\$99.95

Check or M.O.

(201) 329-2770 PHONE ORDERS

Overseas Add \$20.00
N.J. Add 5% Tax
Visa/Mastercard Accepted

DEALER
INQUIRIES
INVITED

*Uniflex is a trademark of Technical Systems Consultants

NEW 6809 SYSTEM!

Now, for about the same price as you would expect to pay for the memory capacity alone, you can have a complete single board computer with these features:

- *6809 CPU, 1MHz clock
- *192KB RAM included, sockets for 64KB more
- *84X24 display of a 7X12 character font
- *Keyboard interface for an un-encoded switch matrix
- *Floppy controller for two 5" drives, single or double sided
- *Parallel printer port
- *Serial I/O port
- *General purpose 8-bit parallel I/O port
- *Parallel expansion port
- *Dimensions: 8.6 by 10.3 inches

NEW FEATURE: Pseudo-disk with 112KB capacity. Now you can use RAM to increase the performance of programs that use disk files.

Assembly programming is optional and is not required for system set-up.

The FLEX operating system is supported by our device drivers. BASIC, PASCAL, and C are available for FLEX. The device drivers (in EPROM) include advanced features like auto-repeat for the keyboard, and track buffering for the disks. Commented source code of all EPROM contents is supplied.

For more information, send a stamped self-addressed envelope and we will send you a configuration guide that explains how to set-up a system. An assembled board is purchased by sending check or money order for \$735 per board. (California add 6% sales tax)

Chandler Microsystems

22051 COSALA
MISSION VIEJO, CA 92691

FLEX, trademark Technical Systems Consultants, Inc.



**SMOKE SIGNAL
BROADCASTING**®

31336 VIA COLINAS

WESTLAKE VILLAGE, CA 91362

TEL. (213) 889-9340 / TWX 910-494-4965

MICROPRODUCT BULLETIN

ARE YOU REALLY SATISFIED?

... with the **PRODUCTS** you are selling?

... with the **PROFITS** you are making?

... with the **SUPPORT** you receive?

Let's face it—

The **PRODUCTS**, the **PROFITS**, and the **SUPPORT** are the key factors in a Dealer-Manufacturer relationship.

IF YOU WANT INCREASED SALES, BIGGER PROFITS, AND REAL SUPPORT — CALL SMOKE SIGNAL BROADCASTING.

We know you've heard that story before. In fact, it probably compelled you to become a dealer in the first place! What you probably **didn't** know is this:

Too many computer companies make ambitious claims when you first express interest, but **very few of them will continue to hold up their end of the deal once you have signed up.**

Smoke Signal **will** deliver what is promised . . . plus much more!

Being a Smoke Signal Broadcasting Dealer is easy, we **don't** require large initial orders; we **don't** impose monthly quotas or contracts on you. What we **do** have are good, reliable, high-quality computer systems; excellent pricing and large discounts. Those factors alone provide enough dealer incentive **without** messy monthly quotas.

Our business computers, the CHIEFTAIN Series, are state-of-the-art 6800/6809 and SS-50 Bus. We run FLEX, OS-9 Multi-user, plus our own DOS68D and DOS69D, with absolutely no changes on your part. Our marketing plans include you, the Dealer — our **real** sales force.

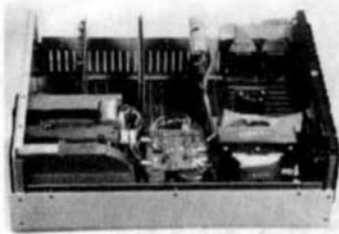
Convince yourself. Call or write Deborah Conrad (213) 889-9340 for more Dealer and OEM information.



FLEX - OS-9 LEVEL ONE - UNIFLEX - OS-9 LEVEL TWO

ONLY GIMIX Systems can be configured to run any of these.

GIMIX systems utilize the most powerful 6809 operating systems: FLEX, UniFLEX, OS-9 LEVEL ONE and TWO -- the systems the PROs use. This means a wide selection of software to choose from as well the ability to develop sophisticated, multi-user/multi-tasking programs on your GIMIX System.



The GIMIX CLASSY CHASSIS™ consists of a heavy-weight aluminum mainframe cabinet which provides more than ample protection for the electronics and 1 or 2 optional 5¼" drives.

Backpanel connectors can be added for convenient connection of terminals, printers, drives and other peripherals.

A 3 position locking keyswitch enables users to disable the front panel reset button to prevent accidental or unauthorized tampering with the system.

The GIMIX system mother board provides fifteen 50 pin slots and eight 30 pin I/O slots -- the most room for expansion of any SS50 system available. The on board baud rate

generator features 11 standard baud rates, 75 to 38.4K, for maximum versatility and compatibility with other systems. Extended address decoding allows the I/O block to be addressed anywhere in the 1 megabyte address space. All components feature Gold plated connectors for a lifetime of solid connections. All boards are fully buffered for maximum system expansion.

Each GIMIX Mainframe System is equipped with an industrial quality power supply featuring a ferro-resonant constant voltage transformer to insure against problems caused by adverse power input conditions such as A.C. line voltage fluctuations etc. The supply provides 8 volts at 30 amps and plus or minus 16 volts at 5 amps, more than enough capacity to power a fully loaded system and two internal drives.

The 2MHz GIMIX 6809 PLUS CPU board includes a time of day clock with battery back-up and 6840 programmable timer to provide the programmer with convenient, accurate time reference. Later addition of 9511 or 9512 arithmetic processors is provided for on the board. The unique GIMIX design enables software selection of either OS-9 or FLEX, both included in many complete GIMIX systems.

GIMIX STATIC RAM boards require no complicated refresh timing cycles or clocks for data retention. GIMIX memory boards are guaranteed for 2 MHz operation with no wait state or clock stretching required.

Our low power NMOS RAM requires less than 3/4 amp at 8V for a fully populated 64K board. For critical situations, our non-volatile 64K byte CMOS static RAM boards with built in battery back-up retain data even with system power removed. A fully charged battery will power this board for a minimum of 21 days. A write protect switch permits CMOS boards to be used for PROM/ROM emulation and software debugging.

The GIMIX DMA controller leaves the processor free to perform other tasks during disk transfers -- an important feature for multi-user/multi-tasking systems where processor time allocation is critical. The DMA board will accommodate up to 4 drives 5¼" or 8" in any combination running single or double density single or double headed. Programmed I/O Disk Controllers are also available.

GIMIX systems are designed with ultimate RELIABILITY in mind. You can choose from the below featured systems or select from our wide variety of components to build a custom package to suit your needs.

GIMIX 2MHz 6809 System including: CLASSY CHASSIS, 6809 PLUS CPU BOARD, 56KB STATIC RAM, 2 SERIAL PORTS W/CABLES, GMXBUG MONITOR, FLEX, and OS-9 LEVEL 1 **\$3248.49**

FOR TWO 5¼" 40 TRACK DSDD DRIVES ADD **\$ 900.00**

GIMIX 128KB WINCHESTER SYSTEM including: CLASSY CHASSIS, 6809 PLUS CPU BOARD, 128KB STATIC RAM, 4 SERIAL PORTS W/CABLES, 5¼" 80 TRACK DSDD FLOPPY DISK DRIVE, 19MB 5¼" WINCHESTER HARD DISK, OS9 LEVEL 2, EDITOR AND ASSEMBLER **\$8998.09**

50HZ Versions Available, 8" Drives Available — Contact GIMIX for Prices and Information.

The Sun Never Sets On A GIMIX!

GIMIX users are found on every continent, including Antarctica. A representative group of GIMIX users includes: Government Research and Scientific Organizations in Australia, Canada, U.K. and in the U.S.; NASA, Oak Ridge, White Plains, Fermilab, Argonne, Scripps, Sloan Kettering, Los Alamos National Labs, AURA, Universities: Carleton, Waterloo, Royal Military College, in Canada; Trier in Germany; and in the U.S.; Stanford, SUNY, Harvard, UCSD, Mississippi, Georgia Tech. Industrial users in Hong Kong, Malaysia, South Africa, Germany, Sweden, and in the U.S.; GTE, Becton Dickinson, American Hoechst, Monsanto, Allied, Honeywell, Perkin Elmer, Johnson Controls, Associated Press, Aydin, Newkirk Electric, Revere Sugar, HI-G/AMS Controls, Chevron, Computer mainframe and peripheral manufacturers, IBM, OKI, Computer Peripherals Inc., Oume, Floating Point Systems, Software houses; Microware, T.S.C., Lucidata, Norpak, Talbot, Stylo Systems, AAA, HMM, Frank Hogg Labs, Epslein Associates, Softwest, Dynasoft, Research Resources U.K., Microworks, Meta Lab, Computerized Business Systems.



GIMIX Inc. reserves the right to change pricing and product specifications at any time without further notice.

GMX is a trademark of GIMIX Inc.

GIMIX® and GHOST® are registered trademarks of GIMIX Inc.

FLEX and UniFLEX are trademarks of Technical Systems Consultants Inc.

OS-9 is a trademark of Microware Inc.

1337 WEST 37th PLACE
CHICAGO, ILLINOIS 60609
(312) 927-5510
TWX 910-221-4055

GIMIX Inc.

The Company that delivers

Quality Electronic products since 1975.

© 1982 GIMIX Inc.
'68' Micro Journal

FLEX & RS COLOR COMPUTER

If you are tired of playing games on your TRS-80C Color Computer, or find that you are handicapped by the limitations of the RS BASIC in trying to write a Program that will allow you to actually USE the Color Computer as a COMPUTER; and if you have been studying the Advertisements in this Magazine and wishing that you could run THESE Programs on your Computer, YOU ARE READY TO MOVE UP TO THE FLEX9 Operating System. If you want to have REAL PROGRAMMING POWER, using an Extremely Powerful Business BASIC, PASCALS, C Compilers, a full-blown Macro Assembler with a Library capability so you are not continuously "reinventing the wheel", YOU ARE READY TO MOVE UP TO THE FLEX9 Operating System. If you would like to see if YOU REALLY COULD USE A COMPUTER IN YOUR BUSINESS, or begin to make your Computer start PAYING ITS OWN WAY by doing some Computer Work for the millions of small businesses around you, such as Wordprocessing, Payroll, Accounting, Inventory, etc., then YOU ARE READY TO MOVE UP TO THE FLEX9 Operating System. How?? DATA-COMP has the way!

DATA-COMP's FLEX9 Conversion for the TRS-80C Color Computer was designed for the SERIOUS COMPUTER USER: with features like greatly increased Display Screens, WITH Lower Case Letters, so you can put a FULL Menu on ONE Screen, or see SEVERAL Paragraphs at the same time; with features like providing a FULL Keyboard so you have FULL Control of your Computer AND its Programs NATURALLY, without needing a chart to see what Key Combination will give you what function; with USER ORIENTED functions to make using the Operating System natural, like having the Computer AUTOMATICALLY determine what type of Disk is being used in what type of Disk Drive and working accordingly, rather than you have to specify each and every thing for it, or like having the Computer work with the Printer you have been using all along without you having to tell the new Operating System what is there; etc., etc., etc.

DATA-COMP has everything you need to make your TRS-80C Color Computer WORK for YOU: from Parts and Pieces to Full, Ready to Use SYSTEMS. DATA-COMP designs, sells, services, and SUPPORTS Computer SYSTEMS, not just Software. CALL DATA-COMP TODAY to make your Computer WORK FOR YOU!

SYSTEM REQUIREMENTS

FLEX9 Special General Version w/Editor & Assembler (which normally sell for \$50.00 ea.) \$150.00
F-MATE(RS) FLEX9 Conversion Rout. for the RS Disk Controller when purchased with Special General FLEX9 Sys. \$49.95
when purchased without the General FLEX9 Sys. \$59.95
NEW -- Full Source Code for the Conv. Routines \$159.95
Set of Eight 64K RAM Chips w/ Mod. Instructions \$99.95
Color Computer with 64K RAM and EXT. BASIC \$549.95
Color Computer with 16K RAM \$375.95
Color Computer with 16K RAM and EXT. BASIC \$465.95

SPECIAL SYSTEM PACKAGES

64K Radio Shack COLOR COMPUTER, Radio Shack COLOR DISK CONTROLLER, a Disk Drive System, Special General Version of FLEX9, F-MATE(RS), and a Box of 10 Double Density Diskettes; a COMPLETE, ready to run SYSTEM on your Color TV Set. \$1249.95

FOR USERS THAT ALREADY HAVE FLEX9 & Disk Drives

Radio Shack DISK CONTROLLER with F-MATE(RS) and a Special Two Drive DISK CABLE \$254.95

DISK DRIVE PACKAGES, with RS Controller

These Packages include the Radio Shack Disk Controller, Disk Drives with Power Supply and Cabinet, and Disk Drive Cable:
PAK #1 ==> 1 Single Sided, Double Density Sys. \$499.95
PAK #2 ==> 2 Single Sided, Double Density Sys. \$729.95
PAK #3 ==> 1 Double Sided, Double Density Sys. \$579.95
PAK #4 ==> 2 Double Sided, Double Density Sys. \$889.95

PARTS AND PIECES

Radio Shack Disk Controller \$169.95
1 ea. Single Sided, Double Density Disk Drive \$249.95
1 ea. Double Sided, Double Density Disk Drive \$349.95
Single Drive Cabinet with Power Supply \$79.95
Double Drive Cabinet with Power Supply \$99.95
Single Drive Disk Cable for RS Controller \$24.95
Double Drive Disk Cable for RS Controller \$34.95
Micro Tech. Prods., Inc. LOWER CASE ROM Adapter \$74.95
Radio Shack BASIC Version 1.1 ROM \$34.95

SOFTWARE



Requires FLEX9 and one of the following CRT terminals

Now Runs On Any Type Terminal

Features:

- Two display boards.
- Four levels of play.
- Point scoring system.
- Play white or black.
- Change or set-up board/piece positions.
- Foxtell move.
- Swap sides.
- Make move and swap sides.
- Change skill level.
- Stop and restart game.
- Solve 'Mate in 1-2-3-4' moves.

\$79.95 Specify 5" or 8" disk

This is one of the strongest CHESS programs running on any microcomputer. estimated USCF Rating 1600

Note: Personal checks allow 3-4 week delivery.

DIET-TRAC Forecaster

A Diet Planning and Analysis Program

DIET-TRAC Forecaster is a program that plans a diet in terms of either calories and percentage of carbohydrates, proteins and fats (C:P:F %) or grams of Carbohydrate, Protein and Fat food exchanges of each of the six basic food groups (vegetable, bread, meat, skim milk, fruit and fat) for a specific individual.

Sex, Age, Height, Present Weight, Frame Size, Activity Level and Basal Metabolic Rate for normal individuals are taken into account. Ideal weight and sustaining calories for any weight of the above individual are calculated. When a weight goal is given (either gain or loss), and a calorie plan is agreed upon between the computer and the individual, the number of days to reach the weight goal is projected. The starting and ending rate of weight loss is calculated, and a daily calendar with each day's predicted weight for a 30-day period is printed.

FLEX VERSION \$59.95
UNIFLEX VERSION \$89.95

PRINTERS

The Epson MX-80

\$455.00

The Epson MX-100

\$725.00

MX-70 \$355.00 MX-80 FT \$575.00

MEMORY

SWTPC-Motorola, MP32

32K Dynamic Memory Board

Assembled & Tested

1 MHZ - No extended addressing

Can be set up for \$0.7FFF or 8000-FFFF

\$149.95



DATA-COMP
SOUTH EAST MEDIA
P.O. Box 794 HIXSON, TN 37343
1-615-842-4601

Verbatim Diskettes.

5" Soft Sector Disks
Single Side Single Density \$2.75 ea.
Single Side Double Density \$2.75 ea.
Double Side Double Density \$4.92 ea.
Plastic Storage Box \$2.00 ea.

8" Soft Sector Disks
Single Side Single Density \$3.75 ea.
Single Side Double Density \$4.18 ea.
Double Side Double Density \$4.75 ea.
Plastic Library Box \$5.00 ea.

Foreign Orders Add 10% Surface—20% Air Mail

DRIVES & CABINETS W/PS

5 1/4"

TANDON - Single Sided, Double Density, 40 Track.....\$249.95

TANDON - Double Sided, Double Density, 40 Track.....\$349.95

CABINET - Single Drive with Power Supply.....\$ 79.95

CABINET - Double Drive with Power Supply.....\$ 99.95

CABLE - Single Drive.....\$ 24.95

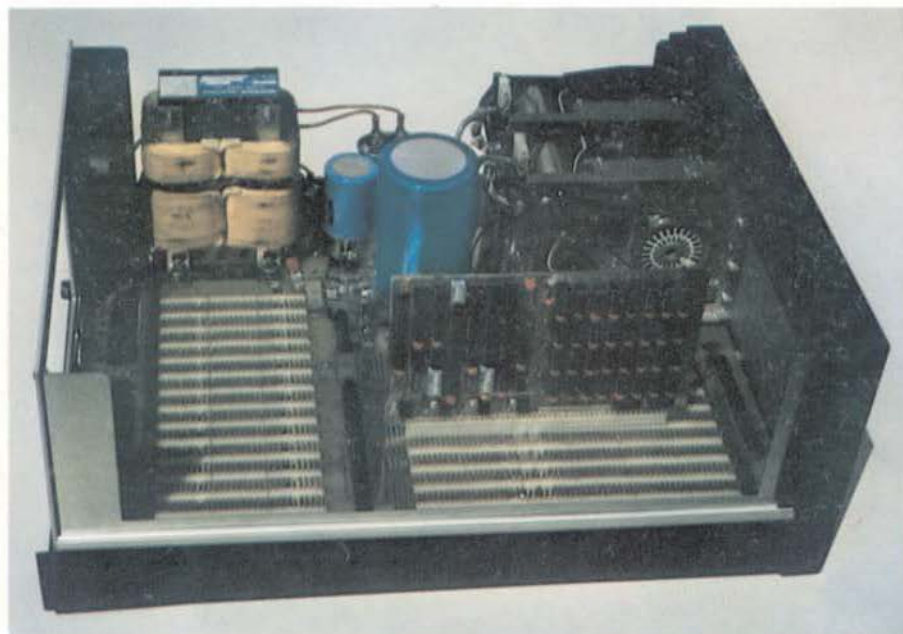
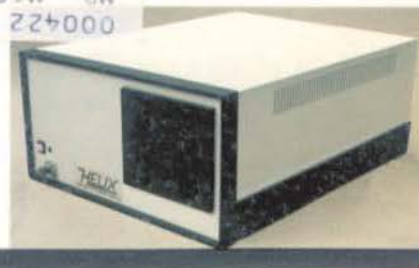
CABLE - Double Drive.....\$ 34.95

NOTE - When ordering cables please specify S50 Bus or Other!!!

Call or write for disk controller Board information.

000422 A/E
MR. MICKEY FERGUSON
P.O. BOX 541
WHITE BLUFF
TN 37187

HELIX™



THE MAINFRAME

- Industry Standard Optima™ Cabinet
- Largest Constant Voltage Power Supply in the industry
- S-64 Bus gives 16 Bit Power and S-50 Bus Compatibility
- 10 Main (S-64) Slots
- 14 I/O (S-30) Slots plus 2 On-board
- On-board Baud Rate Generator to 38.4Kb
- Space and Power for two 5 1/4" Disk Drives
- Full Address Decoding for I/O Slots
- Two RS-232 Serial and Two parallel Ports On-board
- Single Board Construction for Reliability
- Faraday Shielded Bus Lines give "Text Book Clean" Signals

THE PROCESSORS

- 6809**
- Standard 2 MHz Operation
 - Standard DAT Compatible with GIMIX and SWTPC
 - Standard 6840 Interval Timer
 - Standard 1K Scratchpad RAM
 - Standard Clock/Calendar with Battery
 - Provision for Programmers Console
- 68000**
- Standard 8 MHz Operation
 - Memory Management Hardware
 - Provision for Programmers Console
 - 16 Bit Power and 8 Bit Compatibility



THE POWER SUPPLY

- Ferro-resonant Transformer for Line Noise and Under-Voltage Protection
- Conservative 25 Amps at 8.5 Volts
- Conservative 5 Amps at ± 16 Volts
- Conservative Component Rating for Reliability

THE COMPONENTS

- Fully Socketed
- Gold Plated Bus Connectors
- Only "B" Series 68XX Components Used
- Only Top Grade Logic Circuits Used
- Industrial Grade Components Throughout

The HELIX™ computer system represents the latest advance in S-50 bus computer systems. Relying on the physical nature of S-50 bus connectors to guarantee compatibility, the HELIX adds 14 bus lines (becoming S-64) to allow a 68000 processor to operate with full 16 bit data transfer and 24 bit addressing, while at the same time providing full interchangeability with existing S-50 components.

Offered with a selection of processors, memories, and peripheral controllers, a HELIX system can be configured for applications ranging from advanced hobbyist to multiterminal time-sharing.

Designed to offer the utmost in speed, reliability, and utility at a reasonable price, it represents a new standard of quality for those who require a professionally designed computer for professional use.

THE MEMORIES

- DM-64**
- Field Proven
 - Proprietary Memory Control Logic
 - Fully Transparent Refresh
 - Tested at 2.5 MHz Operation
- DM-512**
- 512K Bytes on a Single S-64 Board
 - 16 Bit Power and 8 Bit Compatibility
 - Runs in Existing S-50 Systems where Physical Space Allows
 - Full 24 Bit Addressing
 - Fully Transparent Refresh

THE PRICES

Because of the variety of configurations possible, full pricing cannot be given. Representative prices are:

- 64K 6809 HELIX \$1995
- 64K 68000 HELIX \$2595
- 512K 6809 HELIX \$4450
- 512K 68000 HELIX \$4995

HAZELWOOD COMPUTER SYSTEMS

907 E. Terra, O'Fallon, Missouri 63366 (314) 281-1055

Dealer and OEM Inquiries Invited, We support our Dealers.

Optima is a Trademark of Scientific-Atlanta